Nonlinear State Estimation and Noise Adaptive Kalman Filter Design for Wind Turbines

Vom Fachbereich
Elektrotechnik und Informationstechnik
der Technischen Universität Darmstadt
zur Erlangung des Grades eines Doktor-Ingenieurs (Dr.-Ing.)
genehmigte Dissertation

von

Bastian Ritter, M.Sc.

geboren in Bielefeld

Referent:	Prof. Dr.-Ing. Ulrich Konigorski
Korreferent:	Prof. Dr. Carlo L. Bottasso
Tag der Einreichung:	23. Januar 2020
Tag der Prüfung:	28. April 2020

D 17 · Darmstadt 2020

Nonlinear State Estimation and Noise Adaptive Kalman Filter Design for Wind Turbines

Approved Dissertation by Bastian Ritter, M.Sc.

D 17 · Darmstadt 2020

Printed Version

Available from bookshops or via:
ePubli GmbH, Berlin, Germany
ISBN 978-3-752965-45-2

Electronic Version

Provided by tuprints, E-Publishing-Service der TU Darmstadt
http://tuprints.ulb.tu-darmstadt.de
tuprints@ulb.tu-darmstadt.de

Please refer to this document as:
URN: urn:nbn:de:tuda-tuprints-117852
URL: http://tuprints.ulb.tu-darmstadt.de/11785

Imprint

Bastian Ritter, c/o AutorenServices.de, Birkenallee 24, 36037 Fulda, Germany

Preface

This dissertation has evolved from my work as doctoral student of the Control Systems & Mechatronics Lab at the Technical University of Darmstadt and equally from my work as development engineer at the consulting company Industrial Science GmbH powered by IAV. The research project has unfolded in a trustful collaboration with the IAV GmbH.

First, I would like to express my gratitude to my supervisor Prof. Dr.-Ing. Ulrich Konigorski for making this research possible, for his confidence and for providing a great amount of freedom for my work. Moreover, I thank Prof. Dr. Carlo Bottasso for his kind willingness to accept the review of my thesis and for his effort to evaluate it.

Due to the special character of my research project, I had the privilege to work closely together with many colleagues from the above mentioned industrial partners. First to mention here is Axel with whom I had many inspiring and challenging discussions over the last years which definitely led my way. Moreover, I thank Thorsten for the collaboration and the discussions which contributed certainly to the success of this thesis. Equally, I am very grateful to Mike and Christian who initiated the project and supported me for several years and also during hard times. I thank my colleagues Thibaut, Qin, Jamel, Christian and Mehdi for the open-minded and pleasant working atmosphere. At the same time I appreciated particularly the fruitful and successful collaboration with Michael, Saif, Rafał, Ole, Björn, André, Dang, Erik, Philipp, Axel and Bennet in order to bring forward the industrialization of the real-time supervisory control for wind turbines. Looking back, many aspects of my work revealed an added value for this project which is rewarding in itself. Furthermore, I mention gratefully my students Edwin, Marcus, Thomas, Behsad, Niko, Admir, Janina, Ralf, Roman and Imanol who contributed with major commitment to this research.

I appreciated also very much the shared time with my colleagues at the Control Systems & Mechatronics Lab and the really great *Doktorhut* being a reminder of the past years.

Moreover, I thank my colleagues and friends Dieter, Nicolai, Jonathan, Matthias, Daniel, Jamel, Svenja, Thomas and Lars for their continuous support, for proof-reading as well as for the many technical, political and philosophical conversations.

With sincere gratitude, I mention my four siblings for their continuous support! I thank my parents for paving my way through a good education and supporting my studies for several years. And I thank my parents-in-law for their great encouragement and appreciation.

Most of all and most importantly, I thank my wonderful wife for the unconditional support and my three wonderful children for their joy and boundless energy which both contributed eventually to the completion of this piece of work.

Darmstadt on May 28th, 2020 Bastian Ritter

Dedicated to

Rudi Borgert & Dr. Wolfgang Ritter

Abstract

Modern wind turbines have grown significantly over the last three decades in nacelle heights and rotor diameters and will probably do so in the future. Additionally, more and more turbines are mounted in shallow waters on offshore monopiles and also on floating platforms in the deep sea. These technological developments implicate that the complex structural composition of the turbines becomes more flexible and simultaneously more and competing control objectives appear above the horizon. In this context, advanced state-feedback based control schemes have emerged in the wind sector in order to tackle these new challenges effectively though requiring the mostly hidden and immeasurable information about the dynamic state of the wind turbine.

In order to obtain this valuable information without additional measurements and sensors, the present thesis bridges the scientific gap between the nonlinear estimation theory on the one hand and the practical application to wind turbine control systems on the other hand. This approach includes the investigation of the nonlinear filter algorithms, the control-oriented physical models and the design methodology needed to make nonlinear state estimation techniques ready for wind turbine application. The results of this approach are so-called virtual (model-based) sensors that are employed for multiple estimation tasks, such as the observation of unknown disturbances and the online reconstruction of mechanical component loads. These sensors are applicable whenever it is impossible or too expensive to measure the desired quantities directly.

The thesis explores first the suitable nonlinear algorithms to solve the estimation problems. The focus is here laid upon the sigma-point Kalman filters (SPKF) where classical and adaptive versions are presented. As widely known, the free design parameters of these filters have a significant influence on the expected estimation accuracy. An unfortunate filter parameter design leads to weak filter performance or (even worse) filter divergence. This is very critical in closed-loop systems where the state estimates are essential for the feedback controller. For this reason, the dissertation investigates two approaches to address this problem. The first step is the optimal design of the filter parameters based on numerical optimization. Therewith, all the relevant information about the wind turbine are exploited prior to application in order to find the best initial design parameters. The second step is the selective noise adaptation of certain filter parameters. This approach improves on the filter performance when the previous knowledge is insufficient for a proper initial design or some critical filter parameters are unknown and time-varying.

Moreover, a comprehensive engineering suite is developed in order to integrate the necessary functionality to perform an automated filter performance assessment. Finally, the strengths of these techniques are demonstrated illustratively for a variety of test scenarios, with exten-

sive simulation results, for different estimation problems and different filter types in order to investigate all relevant practical aspects.

In a nutshell, this thesis provides the theoretical foundations, the practical application and also the simulative proof of concept in order to realize the wind turbine state estimation effectively and to bring it successfully into practice. Thereby, these virtual sensors shall level the ground not only for advanced state-feedback control, but also provide further insight into the system's internal behavior which can be exploited in future, for instance, for remaining useful life-time assessment based on reconstructed, experienced loads.

Kurzfassung

Moderne Windturbinen haben in den vergangenen drei Jahrzehnten eines enormes Größenwachstum erfahren, welches sich voraussichtlich auch in der Zukunft noch fortsetzen wird. Weiterhin werden immer mehr Turbinen sowohl in flachen Gewässern und als auch auf schwimmenden Plattformen in der Tiefsee betrieben. Diese technologischen Entwicklungen bringen es mit sich, dass der strukturelle Aufbau von Windturbinen immer elastischer wird und gleichzeitig auch noch mehr Regelziele erfüllt werden müssen, die sich gegenseitig beeinflussen. Erweiterte, modellbasierte Zustandsregler erlauben es diesen neuen Herausforderungen zu begegnen, benötigen aber den vollständigen Zustand der Anlage, welcher im Allgemeinen nicht messbar ist.

Um diese Information ohne zusätzliche Messungen und Sensoren zu erlangen, schließt die vorliegende Arbeit die wissenschaftliche Lücke zwischen der Theorie der nichtlinearen Zustandsschätzung und der praktischen Anwendung für Windturbinen. Dieser Ansatz umfasst die Untersuchung nichtlinearer Filteralgorithmen, der regelungstauglichen Modellierung sowie der Entwurfsmethodik, mit dem Ziel nichtlineare Schätzer für den praktischen Einsatz in geschlossenen Regelkreisen von Windturbinen vorzubereiten.

Das Ergebnis dieses Ansatzes sind sogenannte virtuelle (modellbasierte) Sensoren, die für eine Vielzahl von Schätzaufgaben eingesetzt werden können, wie beispielsweise für die Beobachtung von unbekannten Störgrößen oder die Rekonstruktion von nicht gemessenen mechanischen Lasten. Virtuelle Sensoren sind genau dann von Vorteil, wenn die interessierenden Größen nicht gemessen werden können oder aber dies zu kostenintensiv oder zu aufwändig ist.

Die Arbeit untersucht zunächst die geeigneten nichtlinearen Algorithmen, um die definierten Schätzprobleme zu lösen. Der Fokus liegt hierbei auf den sogenannten Sigma-Punkt Kalman Filtern (SPKF), wobei sowohl klassische als auch adaptive Ansätze berücksichtigt werden. Es ist bekannt, dass die freien Entwurfsparameter des Kalman Filters nachweislich einen großen Einfluss auf die erwartbare Schätzgüte haben. Eine ungünstige Wahl dieser Filterparameter führt zu einer schwachen Performance im Betrieb oder im schlimmsten Fall sogar zur Filterdivergenz. Dies ist insbesondere im geschlossenen Regelkreis kritisch, denn dort liefert die Zustandsschätzung die erforderlichen Informationen für den Regler. Aus diesem Grund werden in dieser Arbeit zwei Ansätze untersucht, um die Anforderung an die Robustheit des Zustandsschätzers für Windenergieanlagen praktisch zu gewährleisten. Zuerst wird der optimierungsbasierte Entwurf mittels numerischer Algorithmen vorgestellt. Dabei wird alles Vorwissen über die Dynamik der Windenergieanlage ausgenutzt, um die besten Startparameter zu finden. Als zweiten Schritt wird die ausgewählte Adaptation einzelner Rauschparameter eingeführt. Dies verbessert die Performance des SPKF, wenn das Vor-

wissen über Prozess- oder Messrauschen unzureichend ist oder kritische Rauschparameter unbekannt und zeitveränderlich sind.

Weiterhin ist eine umfangreiche Entwicklungsumgebung geschaffen worden, die es ermöglicht die erforderlichen Funktionalitäten der Schätzer automatisiert zu bewerten. Schließlich werden die Stärken der klassischen und adaptiven Filter in umfangreichen Simulationsstudien verglichen, um alle praktisch relevanten Aspekte mit in die Untersuchungen einzubeziehen.

Zusammenfassend liefert diese Dissertation das theoretische Fundament, die praktische Realisierung und den simulativen Funktionsnachweis, um virtuelle Sensoren effektiv und erfolgreich praktisch umzusetzen. Damit sollen es diese modellbasierten Sensoren ermöglichen zukünftige zustandbasierte Regelung von Windturbinen zu flankieren und darüber hinaus tiefergehenden Einblick in das interne Verhalten des Systems zu erlangen, was zukünftig beispielweise für die Verbesserung der Schätzung bzw. Vorhersage der verbleibenden Lebensdauer von Komponenten genutzt werden kann.

Contents

Symbols and Acronyms

Roman Symbols

$\mathbf{A}_k$	$\in \mathbb{R}^{n_x \times n_x}$	discrete-time system matrix	
A	$\in \mathbb{R}^{+}$	rotor swept area	m^2
$\mathcal{A}(\varphi)$	$\in \mathbb{R}^{n_{\mathrm{bl}} \times n_{\mathrm{bl}}}$	tower dam influence matrix	-
a, a_i	$\in \mathbb{R}$	axial induction factor (at blade section i)	-
a', a'_i	$\in \mathbb{R}$	tangential induction factor (at blade section i)	-
$\mathbf{B}_k$	$\in \mathbb{R}^{n_x \times n_u}$	discrete-time control input matrix	
b	$\in \mathbb{Z}^{+}$	b-th rotor blade	-
b_i	$\in \mathbb{R}^{+}$	blade section width	m
b_φ	$\in \mathbb{R}^{+}$	torsional damping coefficient drive-train	Nms/rad
$b_{\mathrm{T}x}$	$\in \mathbb{R}^{+}$	damping coefficient (nacelle fore-aft direction)	Ns/m
$b_{\mathrm{T}y}$	$\in \mathbb{R}^{+}$	damping coefficient (nacelle sidewise direction)	Ns/m
$\mathbf{C}_k$	$\in \mathbb{R}^{n_y \times n_x}$	discrete-time output matrix	
$C_{\mathrm{D}}, C_{\mathrm{D},i}$	$\in \mathbb{R}$	drag coefficient (at blade section i)	-
$C_{\mathrm{L}}, C_{\mathrm{L},i}$	$\in \mathbb{R}$	lift coefficient (at blade section i)	-
C_{M}	$\in \mathbb{R}$	aerodynamic torque coefficient	-
$C_{\mathrm{n}}, C_{\mathrm{n},i}$	$\in \mathbb{R}$	normal force coefficient (at blade section i)	-
C_{P}	$\in \mathbb{R}$	aerodynamic power coefficient	-
$C_{\mathrm{P,max}}$	$\in \mathbb{R}$	maximum aerodynamic power coefficient	-
C_{P}^*	$\in \mathbb{R}$	optimal aerodynamic power coefficient	-
C_{T}	$\in \mathbb{R}$	aerodynamic thrust coefficient	-
$C_{\mathrm{t}}, C_{\mathrm{t},i}$	$\in \mathbb{R}$	tangential force coefficient (at blade section i)	-
c_i	$\in \mathbb{R}^{+}$	blade section chord length	m
$\mathbf{D}_k$	$\in \mathbb{R}^{n_y \times n_u}$	discrete-time direct-feedthrough matrix	
$\boldsymbol{d}, \boldsymbol{d}_k$	$\in \mathbb{R}^{n_d}$	disturbance input vector	
D	$\in \mathbb{R}^{+}$	rotor diameter	m
d_{T}	$\in \mathbb{R}^{+}$	distance from tower center to rotor hub	m
$\boldsymbol{e}, \boldsymbol{e}_k$	$\in \mathbb{R}^{n_x}$	state estimation error	
F_i	$\in \mathbb{R}$	resulting aerodynamic force (at blade section i)	N
$F_{\mathrm{L},i}$	$\in \mathbb{R}$	aerodynamic lift force (at blade section i)	N
$F_{\mathrm{D},i}$	$\in \mathbb{R}$	aerodynamic drag force (at blade section i)	N
$F_{\mathrm{n},b}$	$\in \mathbb{R}$	aerodynamic normal blade force (at blade b)	N
$F_{\mathrm{n},i}^*$	$\in \mathbb{R}$	aerodynamic normal force (at blade section i)	N
F_{S}	$\in \mathbb{R}$	aerodynamic sidewise force	N
F_{T}	$\in \mathbb{R}$	aerodynamic rotor thrust force	N
$F_{\mathrm{t},b}$	$\in \mathbb{R}$	aerodynamic tangential blade force (at blade b)	N
$F_{\mathrm{t},i}^*$	$\in \mathbb{R}$	aerodynamic tangential force (at blade section i)	N

$\boldsymbol{f}(\cdot)$	$\in \mathbb{R}^{n_x}$	continuous-time process model	
$\boldsymbol{f}_k(\cdot)$	$\in \mathbb{R}^{n_x}$	discrete-time process model	
H	$\in \mathbb{R}$	hub height	m
$\boldsymbol{h}(\cdot)$	$\in \mathbb{R}^{n_y}$	continuous-time output model	
$\boldsymbol{h}_k(\cdot)$	$\in \mathbb{R}^{n_y}$	discrete-time output model	
i_{gb}	$\in \mathbb{R}^{+}$	drive-train gear-box ratio	-
$J(\boldsymbol{z})$	$\in \mathbb{R}^{+}$	cost function depending on decision variables $\boldsymbol{z}$	-
J_{NS}	$\in \mathbb{R}^{+}$	cost function based on NEES and NIS	-
J_{WS}	$\in \mathbb{R}^{+}$	cost function based on WEES and WIS	-
$\boldsymbol{\mathcal{K}}_k$	$\in \mathbb{R}^{n_x \times n_y}$	Kalman gain matrix	
$\boldsymbol{\mathcal{K}}_k^{\#}$	$\in \mathbb{R}^{n_y \times n_x}$	pseudo inverse of Kalman gain matrix	
k	$\in \mathbb{Z}$	time index $t_k = kT_{\mathrm{s}}$	-
k_{T}	$\in \mathbb{R}^{+}$	first modal stiffness of tower-nacelle-rotor system	N/m
$k_{\mathrm{T}x}$	$\in \mathbb{R}^{+}$	stiffness coefficient in nacelle fore-aft direction	N/m
$k_{\mathrm{T}y}$	$\in \mathbb{R}^{+}$	stiffness coefficient in nacelle side-side direction	N/m
k_{φ}	$\in \mathbb{R}^{+}$	torsional spring coefficient drive-train	Nm/rad
L	$\in \mathbb{R}^{+}$	turbulence length scale	m
$\mathbf{M}_k^{-}$	$\in \mathbb{R}^{n_x \times n_x}$	matrix inverse of $\mathbf{P}_k^{-}$	
$\mathbf{M}_k^{+}$	$\in \mathbb{R}^{n_x \times n_x}$	matrix inverse of $\mathbf{P}_k^{+}$	
$\mathbf{M}_k^{yy}$	$\in \mathbb{R}^{n_y \times n_y}$	matrix inverse of $\mathbf{P}_k^{yy}$	
M	$\in \mathbb{Z}^{+}$	number of samples	-
M_{a}	$\in \mathbb{R}$	aerodynamic rotor torque	Nm
$\hat{M}_{\mathrm{a}}$	$\in \mathbb{R}$	estimated aerodynamic rotor torque	Nm
M_{DT}	$\in \mathbb{R}$	drive-train torque	Nm
M_{g}	$\in \mathbb{R}$	electrical generator torque	Nm
M_{nod}	$\in \mathbb{R}$	aerodynamic nodding moment	Nm
$M_{\mathrm{T}x}$	$\in \mathbb{R}$	sidewise tower base bending moment	Nm
$M_{\mathrm{T}y}$	$\in \mathbb{R}$	axial tower base bending moment	Nm
m_{T}	$\in \mathbb{R}^{+}$	first modal mass of tower-nacelle-rotor system	kg
N	$\in \mathbb{Z}^{+}$	number of samples	-
$\boldsymbol{n}_k$	$\in \mathbb{R}^{n_y}$	vector of measurement noise	
n_{bl}	$\in \mathbb{Z}^{+}$	number of rotor blades	-
n_{sec}	$\in \mathbb{Z}^{+}$	number of blade sections	-
n_{g}	$\in \mathbb{R}^{+}$	generator speed (high-speed side)	rpm
n_{r}	$\in \mathbb{R}^{+}$	rotor speed (low-speed side)	rpm
n_u	$\in \mathbb{Z}^{+}$	number of control inputs (dimension of $\boldsymbol{u}_k$)	-
n_x	$\in \mathbb{Z}^{+}$	number of states (dimension of $\boldsymbol{x}_k$)	-
n_y	$\in \mathbb{Z}^{+}$	number of measured outputs (dimension of $\boldsymbol{y}_k$)	-
n_z	$\in \mathbb{Z}^{+}$	number of performance or decision variables	-
$\mathbf{P}_k^{-}$	$\in \mathbb{R}^{n_x \times n_x}$	a priori state error covariance matrix	
$\mathbf{P}_k^{+}$	$\in \mathbb{R}^{n_x \times n_x}$	posterior state error covariance matrix	
$\widetilde{\mathbf{P}}_k$	$\in \mathbb{R}^{n_x \times n_x}$	filter computed residual covariance matrix	

$\hat{\tilde{\mathbf{P}}}_k$	$\in \mathbb{R}^{n_x \times n_x}$	empirical residual covariance matrix	
$\mathbf{P}_k^{xy}$	$\in \mathbb{R}^{n_x \times n_y}$	error cross-covariance matrix	
$\mathbf{P}_k^{yy}$	$\in \mathbb{R}^{n_y \times n_y}$	filter computed innovation error covariance matrix	
$\hat{\mathbf{P}}_k^{yy}$	$\in \mathbb{R}^{n_y \times n_y}$	empirical innovation error covariance matrix	
P_a	$\in \mathbb{R}$	aerodynamic power at the rotor	W
P_avail	$\in \mathbb{R}$	available aerodynamic power in the wind	W
P_g	$\in \mathbb{R}$	electrical generator power	W
$\boldsymbol{p}_k$	$\in \mathbb{R}^{n_x}$	known model parameters	
$\mathbf{Q}_k$	$\in \mathbb{R}^{n_x \times n_x}$	process noise covariance matrix	
$\boldsymbol{q}_k$	$\in \mathbb{R}^{n_x}$	vector of diagonal elements of $\mathbf{Q}_k$	
$\mathbf{R}_k$	$\in \mathbb{R}^{n_y \times n_y}$	measurement noise covariance matrix	
R	$\in \mathbb{R}$	blade tip radius	m
$\boldsymbol{r}_k$	$\in \mathbb{R}^{n_y}$	vector of diagonal elements of $\mathbf{R}_k$	
r, r_i	$\in \mathbb{R}$	blade radius (at blade section i)	m
r_B	$\in \mathbb{R}$	blade effective radius	m
r_n	$\in \mathbb{R}$	normal effective blade radius	m
r_T	$\in \mathbb{R}$	tower radius	m
r_t	$\in \mathbb{R}$	tangential effective radius	m
$\mathbf{S}_k^-$	$\in \mathbb{R}^{n_x \times n_x}$	matrix square-root of $\mathbf{P}_k^-$	
$\mathbf{S}_k^+$	$\in \mathbb{R}^{n_x \times n_x}$	matrix square-root of $\mathbf{P}_k^+$	
$\mathbf{S}_k^{yy}$	$\in \mathbb{R}^{n_y \times n_y}$	matrix square-root of $\mathbf{P}_k^{yy}$	
$\mathbf{S}_k^\mathrm{Q}$	$\in \mathbb{R}^{n_x \times n_x}$	matrix square-root of $\mathbf{Q}_k$	
$\mathbf{S}_k^\mathrm{R}$	$\in \mathbb{R}^{n_y \times n_y}$	matrix square-root of $\mathbf{R}_k$	
TI	$\in \mathbb{R}^+$	turbulence intensity	-
T_in	$\in \mathbb{R}^+$	wind inflow time constant	s
T_S	$\in \mathbb{R}^+$	sampling time	s
T_w	$\in \mathbb{R}^+$	mean wind time constant	s
t, t_k	$\in \mathbb{R}$	current simulation time	s
$\boldsymbol{u}, \boldsymbol{u}_k$	$\in \mathbb{R}^{n_u}$	control input vector	
$\boldsymbol{v}_k$	$\in \mathbb{R}^{n_y}$	vector of innovations (output prediction error)	
v_0	$\in \mathbb{R}$	0-component of the wind inflow	m/s
v_d	$\in \mathbb{R}$	d-component of the wind inflow	m/s
v_in	$\in \mathbb{R}^+$	inflow wind speed	m/s
v_b	$\in \mathbb{R}$	blade effective wind speed	m/s
v_m	$\in \mathbb{R}^+$	average wind speed	m/s
v_q	$\in \mathbb{R}$	q-component of the wind inflow	m/s
v_R	$\in \mathbb{R}^+$	wind speed in the rotor disk plane	m/s
$v_{\mathrm{r},i}$	$\in \mathbb{R}^+$	relative inflow velocity at blade section i	m/s
v_w, v_∞	$\in \mathbb{R}^+$	rotor effective (free-stream) wind speed	m/s
v_x	$\in \mathbb{R}^+$	x-component of the blade wind speed	m/s
v_y	$\in \mathbb{R}^+$	y-component of the blade wind speed	m/s
$\mathbf{W}_x$	$\in \mathbb{R}^{n_x \times n_x}$	weighting matrix for the state variables	

$\mathbf{W}_{\dot{x}}$	$\in \mathbb{R}^{n_x \times n_x}$	weighting matrix for the state derivatives	
$\mathbf{W}_{y}$	$\in \mathbb{R}^{n_y \times n_y}$	weighting matrix for the output variables	
$\mathbf{W}_{\dot{y}}$	$\in \mathbb{R}^{n_y \times n_y}$	weighting matrix for the output derivatives	
$\boldsymbol{w}_k$	$\in \mathbb{R}^{n_x}$	vector of process noise	
w	$\in [0,1]$	weighting factor in cost function J_{NS}	-
$w_{\mathrm{c},i}$	$\in \mathbb{R}$	UKF parameter for the computation of the covariance	-
w_{m}	$\in \mathbb{R}$	CKF parameter for the means and covariances	-
$w_{\mathrm{m},i}$	$\in \mathbb{R}$	UKF parameter for the computation of the mean	-
w_{p}	$\in \mathbb{R}$	SPKF parameter for the distribution of the SP	-
$\boldsymbol{\mathcal{X}}_{k-1}^{+}$	$\in \mathbb{R}^{n_x \times 2n_x}$	SP before propagation through process model $\boldsymbol{f}_k$	
$\boldsymbol{\mathcal{X}}_{k}^{*}$	$\in \mathbb{R}^{n_x \times 2n_x}$	SP after propagation through process model $\boldsymbol{f}_k$	
$\boldsymbol{\mathcal{X}}_{k}^{-}$	$\in \mathbb{R}^{n_x \times 2n_x}$	SP before propagation through output model $\boldsymbol{h}_k$	
$\widehat{\mathbf{X}}_{k-1}^{+}$	$\in \mathbb{R}^{n_x \times 2n_x}$	matrix with $2n_x$ column vectors of $\hat{\boldsymbol{x}}_{k-1}^{+}$	
$\boldsymbol{x}, \boldsymbol{x}_k$	$\in \mathbb{R}^{n_x}$	state vector (true value)	
$\vec{\boldsymbol{x}}_{\mathrm{B},b}$	$\in \mathbb{R}^{3}$	blade position vector	m
$\hat{\boldsymbol{x}}_{k}^{-}$	$\in \mathbb{R}^{n_x}$	state vector (a priori estimate)	
$\hat{\boldsymbol{x}}_{k}^{+}$	$\in \mathbb{R}^{n_x}$	state vector (posterior estimate)	
$\widetilde{\boldsymbol{x}}_k$	$\in \mathbb{R}^{n_x}$	state residual	
$x_i, x_{i,k}$	$\in \mathbb{R}$	i-th state variable (at time t_k)	
x_{T}	$\in \mathbb{R}$	nacelle position in fore-aft direction	m
$\dot{x}_{\mathrm{T}}$	$\in \mathbb{R}$	nacelle velocity in fore-aft direction	m/s
$\ddot{x}_{\mathrm{T}}$	$\in \mathbb{R}$	nacelle acceleration in fore-aft direction	m/s^2
$\boldsymbol{\mathcal{Y}}_{k}^{*}$	$\in \mathbb{R}^{n_y \times 2n_x}$	SP after propagation through output model $\boldsymbol{h}_k$	
$\widehat{\mathbf{Y}}_{k}$	$\in \mathbb{R}^{n_y \times 2n_x}$	matrix with $2n_x$ column vectors of $\hat{\boldsymbol{y}}_k$	
$\boldsymbol{y}, \boldsymbol{y}_k$	$\in \mathbb{R}^{n_y}$	output vector (observations/measured outputs)	
$\hat{\boldsymbol{y}}_k$	$\in \mathbb{R}^{n_y}$	predicted system output vector	
y_{T}	$\in \mathbb{R}$	nacelle position in side-side direction	m
$\dot{y}_{\mathrm{T}}$	$\in \mathbb{R}$	nacelle velocity in side-side direction	m/s
$\ddot{y}_{\mathrm{T}}$	$\in \mathbb{R}$	nacelle acceleration in side-side direction	m/s^2
$\boldsymbol{z}$	$\in \mathbb{R}^{n_z}$	vector of decision variables (general)	
$\boldsymbol{z}, \boldsymbol{z}_k$	$\in \mathbb{R}^{n_z}$	vector of performance outputs (true values)	
$\hat{\boldsymbol{z}}_k$	$\in \mathbb{R}^{n_z}$	vector of performance outputs (estimated values)	

Greek Symbols

α, α_i	$\in \mathbb{R}$	angle of attack (at blade section i)	deg
α	$\in [0,1]$	scaling parameter for the unscented transformation	-
α_h	$\in \mathbb{R}$	horizontal inflow angle	deg
α_v	$\in \mathbb{R}$	vertical inflow angle	deg
β, β_c	$\in \mathbb{R}$	collective blade pitch angle	deg
β_b	$\in \mathbb{R}$	individual blade pitch angle of blade b	deg
β	$\in \mathbb{R}^+$	UKF/SR-UKF parameter	-
$\gamma, \gamma_\mathrm{w}$	$\in \mathbb{R}$	nacelle yaw angle	deg
γ_b	$\in \mathbb{R}$	tower dam yaw angle of rotor blade b	deg
ϕ, ϕ_i	$\in \mathbb{R}$	inflow angle (at blade section i)	deg
$\Delta\varphi$	$\in \mathbb{R}$	drive-train torsion angle	-
$\Delta\dot{\varphi}$	$\in \mathbb{R}$	drive-train torsional speed	1/s
$\delta_\mathrm{h}, \delta'_\mathrm{h}$	$\in \mathbb{R}$	linear horizontal shear coefficient	-
$\delta_\mathrm{v}, \delta'_\mathrm{v}$	$\in \mathbb{R}$	linear vertical shear coefficient	-
$\varepsilon_{\mathrm{NS},k}$	$\in \mathbb{R}^+$	normalized estimation error squared	-
$\bar{\varepsilon}_\mathrm{NS}$	$\in \mathbb{R}^+$	average normalized estimation error squared	-
$\varepsilon_{\mathrm{WS},k}$	$\in \mathbb{R}^+$	weighted estimation error squared	-
$\bar{\varepsilon}_\mathrm{WS}$	$\in \mathbb{R}^+$	average weighted estimation error squared	-
κ	$\in \mathbb{R}$	vertical shear exponent	-
κ	$\in \mathbb{R}$	UKF parameter	-
φ	$\in \mathbb{R}$	rotor azimuth angle	-
$\dot{\varphi}, \Omega$	$\in \mathbb{R}$	rotor angular speed (low speed side)	1/s
$\dot{\varphi}_\mathrm{g}$	$\in \mathbb{R}$	generator angular speed (low speed side)	1/s
ϱ	$\in \mathbb{R}$	air mass density	$\mathrm{kg/m^3}$
η	$\in [0,1]$	mechanical efficiency	-
η_g	$\in [0,1]$	efficiency of the electrical generator	-
λ	$\in \mathbb{R}$	rotor tip-speed-ratio	-
λ	$\in \mathbb{R}$	UKF/SR-UKF parameter	-
λ_b	$\in \mathbb{R}$	blade tip-speed-ratio	-
λ^*	$\in \mathbb{R}$	optimal tip-speed-ratio	-
λ_r	$\in \mathbb{R}$	local tip-speed-ratio	-
$\boldsymbol{\xi}$	$\in \mathbb{R}^{n_x}$	augmented state vector	
ψ_b	$\in \mathbb{R}$	rotor blade azimuth angle of rotor blade b	deg
ψ_r	$\in \mathbb{R}$	rotor tilt angle	rad
$\boldsymbol{\vartheta}, \boldsymbol{\vartheta}_k$	$\in \mathbb{R}^{n_\theta}$	unknown model parameters (to be estimated)	
θ_i	$\in \mathbb{R}$	local pitch angle at blade section i	deg
σ, σ_i	$\in \mathbb{R}^+$	blade solidity (at blade section i)	-
Θ	$\in \mathbb{R}^+$	equivalent rotor-generator inertia (low-speed side)	$\mathrm{kg\,m^2}$
Θ_g	$\in \mathbb{R}^+$	generator rotational inertia (high-speed side)	$\mathrm{kg\,m^2}$
Θ_r	$\in \mathbb{R}^+$	rotor inertia (low-speed side)	$\mathrm{kg\,m^2}$
χ	$\in [0,1]$	tower dam reduction factor	-

ω_0	$\in \mathbb{R}^+$	tower fore-aft natural frequency	1/s
ω_{DT}	$\in \mathbb{R}^+$	drive-train natural frequency	1/s
ζ	$\in \mathbb{R}^+$	beam coupling coefficient	1/m
ζ_{DT}	$\in \mathbb{R}$	modal damping drive-train	-
$\zeta_{\mathrm{T}x}$	$\in \mathbb{R}$	modal damping (nacelle fore-aft direction)	-
$\zeta_{\mathrm{T}y}$	$\in \mathbb{R}$	modal damping (nacelle side-side direction)	-
$\nu_{\mathrm{NS},k}$	$\in \mathbb{R}^+$	normalized innovation squared	-
$\bar{\nu}_{\mathrm{NS}}$	$\in \mathbb{R}^+$	sum of normalized innovation squared	-
$\nu_{\mathrm{WS},k}$	$\in \mathbb{R}^+$	weighted innovation squared	-
$\bar{\nu}_{\mathrm{WS}}$	$\in \mathbb{R}^+$	sum of weighted innovation squared	-
χ^2	$\in \mathbb{R}^+$	chi-squared distributed/distribution	-
μ_{T}	$\in \mathbb{R}^+$	maximum velocity deficit (tower dam model)	-
σ_{T}	$\in \mathbb{R}^+$	width parameter (tower dam model)	-

Indices and Exponents

$(\cdot)^0, (\cdot)_0$	initial value	$(\cdot)_i$	blade section element
$(\cdot)^{\mathrm{T}}$	transpose of a vector or matrix	$(\cdot)_j$	related to parameter j
$(\cdot)^*$	optimal value (exception transformed sigma-points)	$(\cdot)_k$	variable at time step t_k
		$(\cdot)_l$	variable at time step t_l
$(\cdot)^+$	corrected value	$(\cdot)_{\mathrm{n}}$	normal
$(\cdot)^-$	predicted value	$(\cdot)_{\mathrm{r}}$	related to the rotor
$\bar{(\cdot)}$	upper bound constraints	$(\cdot)_{\mathrm{ref}}$	reference value
$\underline{(\cdot)}$	lower bound constraints	$(\cdot)_{\mathrm{S}}$	variables related to the slave filter
$(\cdot)_{\mathrm{a}}$	aerodynamic	$(\cdot)_{\mathrm{T}}$	related to the rotor thrust
$(\cdot)_b$	related to rotor blade b	$(\cdot)_{\mathrm{T}}$	related to the tower (tower top)
$(\cdot)_{\mathrm{DT}}$	related to the drive-train	$(\cdot)_{\mathrm{t}}$	tangential
$(\cdot)_{\mathrm{d}}$	constant value	$(\cdot)_x$	related to fore-aft direction
$(\cdot)_{\mathrm{g}}$	related to the generator	$(\cdot)_y$	related to side-side direction
$(\cdot)_i, (\cdot)_{k,i}$	related to the i-th state variable		

Acronyms

AEP	Annual Energy Production
ANEES	Average Normalized Estimation Error Squared
ASM	Active Set Method
AWS	Average Wind Speed
BEM	Blade Element Momentum (theory)
CAPEX	Capital Expenditure
CDKF	Central-Difference Kalman Filter
CKF	Cubature Kalman Filter
CPC	Collective Pitch Control
COP	Constrained Optimization Problem
DARE	Discrete-time Algebraic Riccati Eq.
DE	Dual Estimation
DEL	Damage Equivalent Load
DOA	Distributed Observer Architecture
DoE	United States Department of Energy
DSSM	Dynamic State-Space Model
DTU	Technical University of Denmark
EE	Estimation Error
EKF	Extended Kalman Filter
EOG	Empirical Observability Gramian
EPM	Estimator Performance Management
EWS	(Rotor-) Effective Wind Speed
FAST	Fatigue, Aerodynamics, Structure, and Turbulence simulator (by NREL)
FCR	Fixed Charge Rate (for CAPEX)
FFT	Fast Fourier Transform
FLR	Full Load Regime
FPA	Filter Performance Assessment
FOWT	Floating Offshore Wind Turbines
GHKF	Gauss-Hermite Kalman Filter
HAWT	Horizontal Axis Wind Turbine
IAE	Innovation Adaptive Estimation
IEKF	Iterated-Extended Kalman Filter
IMU	Inertial Measurement Unit
IPC	Individual Pitch Control
IPM	Interior-Point Methods
KF	Kalman Filter
KG	Kalman Gain
KPI	Key Performance Indicator
LCoE	Levelized Costs of (electrical) Energy
lidar	light detection and ranging
LKF	Linearized Kalman Filter
LRKF	Linear Regression Kalman Filter
LSE	Least Squares Estimation
MB	Maybeck estimator
MHE	Moving Horizon Estimation
MLE	Maximum Likelihood Estimation
MMAE	Multiple-Model Adaptive Estimation
MOA	Monolithic Observer Architecture
MPC	Model Predictive Control
MPPT	Maximum Power Point Tracking
MS-CKF	Master-Slave Cubature Kalman Filter
MSE	Mean-Squared Error
MSR	Matrix Square-Root
NACA	National Advisory Committee for Aeronautics (today part of NASA)
NEES	Normalized Estimation Error Squared
NIS	Normalized Innovation Squared
NMPC	Nonlinear Model Predictive Control
NMSE	Normalized Mean-Squared Error
NREL	National Renewable Energy Laboratory
NWTC	National Wind Technology Center of the US Dep. of Energy (DoE)
ODE	Ordinary Differential Equations
OP	Optimization Problem
OPEX	Operating Expenditure
PLR	Partial Load Regime
RMSE	Root Mean-Squared Error
RV	Random Variable(s)
SCADA	Supervisory Control and Data Acquisition (System)
SNIS	Sum of Normalized Innovation Squared
SOEKF	Second-order Extended Kalman Filter
SP	Sigma-points ($\boldsymbol{\mathcal{S}_\mathcal{P}}$)
SPKF	Sigma-Point Kalman Filter
SQP	Sequential Quadratic Programming
SR-CKF	Square-Root Cubature Kalman Filter
SR-UKF	Square-Root Unscented Kalman Filter
SSKF	Spherical-Simplex Kalman Filter
SVD	Singular Value Decomposition
UIO	Unknown Input Observer
UKF	Unscented Kalman Filter
UT	Unscented Transform
TI	Turbulence Intensity/Intensities
TSR	Tip-Speed-Ratio
VAWT	Vertical Axis Wind Turbine
VSVP	Variable-Speed Variable-Pitch
WEES	Weighted Estimation Error Squared
WIS	Weighted Innovation Squared
WT	Wind Turbine
WTC	Wind Turbine Control

1 Introduction

This chapter starts with the motivation for the presented research and highlights the niche of this dissertation. It provides additionally a detailed review on the existing literature in the field of state estimation for wind turbines and puts emphasis on the research objectives of this thesis. Finally, the author's contributions are presented and a detailed outline of the thesis is given.

1.1 Motivation

After almost four decades of intensive research of wind turbine technology, wind power plants have grown up to one of the largest energy conversion systems on the planet. Today's modern turbines generate up to 10 MW of electrical power and their rotor area covers more than 20.000 m^2. More and more turbines are installed offshore on bottom-fixed foundations or even floating platforms.

Wind turbine manufacturers and wind energy scientists from various fields of research have improved component and turbine design to meet the steadily increasing requirements for reliable renewable energy generating plants. Substantial progress in aerodynamic blade design, understanding of complex wind evolution, light-weight composite materials, (remote) sensing devices and load mitigating control strategies, among others, has made this development possible. Therewith the wind energy sector has become the most promising sector for future clean and secure energy supply. It exhibits in addition a small environmental impact in comparison to other fossil fuel-based and often detrimental technologies.

Yet, the competition from long-established energy sectors remains relentlessly strong and the levelized costs of energy (LCoE) must drop further to address and tackle the economic requirements (see Fig. 1.1). The LCoE is defined by

$$\mathrm{LCoE} = \frac{\mathrm{OPEX} + \mathrm{CAPEX} \cdot \mathrm{FCR}}{\mathrm{AEP}} \tag{1.1}$$

which is the ratio of the lifetime costs per year divided by the profit from the annual energy production (AEP) [45, 137, 91]. The lifetime costs consist of the annual operational expenditure (OPEX), the installed capital expenditure (CAPEX) and the fixed charge rate (FCR). Therefore, to make wind energy more cost-effective either the AEP must be increased or the lifetime costs must drop. For this purpose, the research community must continue to find innovative solutions for the most challenging topics in wind energy science.

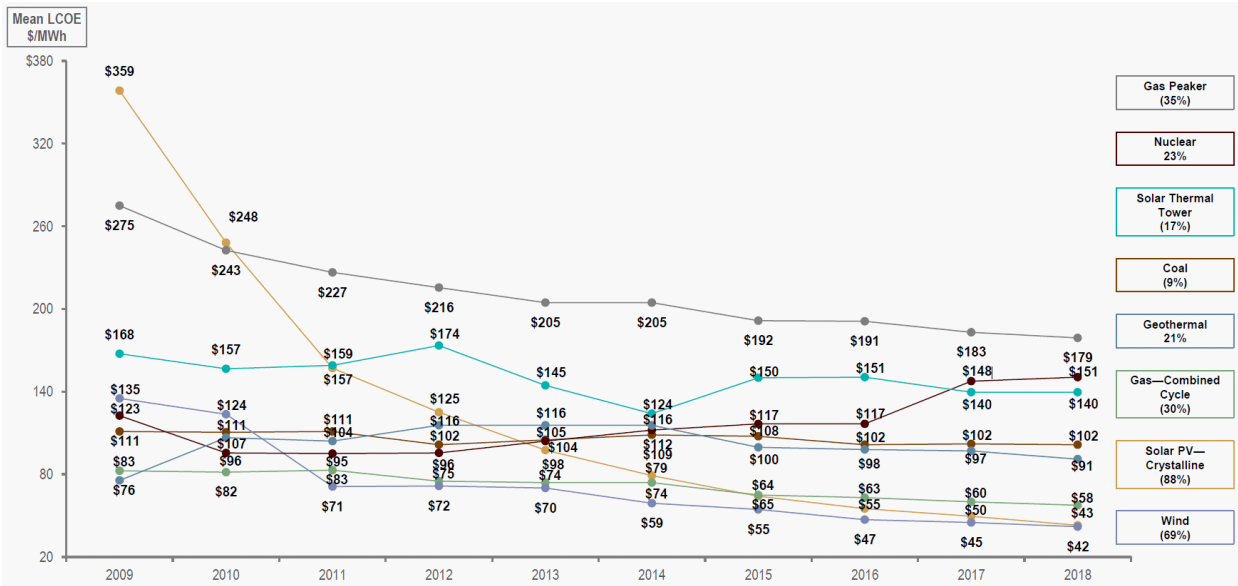

Figure 1.1: Levelized cost of energy comparison: Historical utility-scale generation comparison with selected historical mean unsubsidized LCoE values. Source: Lazard 2018, cf. [115] (p. 7)

One of these relevant fields of research is the theory of control systems which is more important than ever for advanced wind turbine applications. It unifies the theories of dynamic systems, automatic and feedback-control as well as state estimation.

Since the millennium the growth of wind turbines and the necessity for lowering the LCoE has triggered a lot of innovative research for active vibration and individual pitch control (IPC) to reduce fatigue loads on costly and valuable components [20]. These active strategies use the existing control devices to mitigate mechanical loads on the tower, the blades and drive-train instead of over-dimensioning of components (both influencing the CAPEX and OPEX). These goals have become relevant objectives for wind turbine control [17] (p. 52/53).

Yet, these load-oriented objectives compete with the fundamental requirement of every modern wind turbine generator to maximize the energy harvest (AEP). The closed-loop wind turbine controller is therefore crucial for both fatigue load reduction and reliable maximum power production. As a consequence, wind turbines have seen an increasing complexity for the control designers to achieve all control objectives at a time, although some simplifications are possible [23]. In order to tackle the increasing interactions with the growth of wind turbines and to reduce also the expense for the controller design, advanced control schemes and advanced design methodologies come into play (cf. Fig. 1.2). Among them, nonlinear model predictive control (NMPC) has emerged as a promising and strong candidate for future wind turbine control (WTC) [187, 110, 183]. The main reasons for that are the capability to handle complex operational limitations and to incorporate the preview information about the disturbance inputs (wind speed and shear) elegantly [60]. Insofar implemented efficiently, wind turbine NMPC runs in real-time on state-of-the-art industrial controllers, cf. [183].

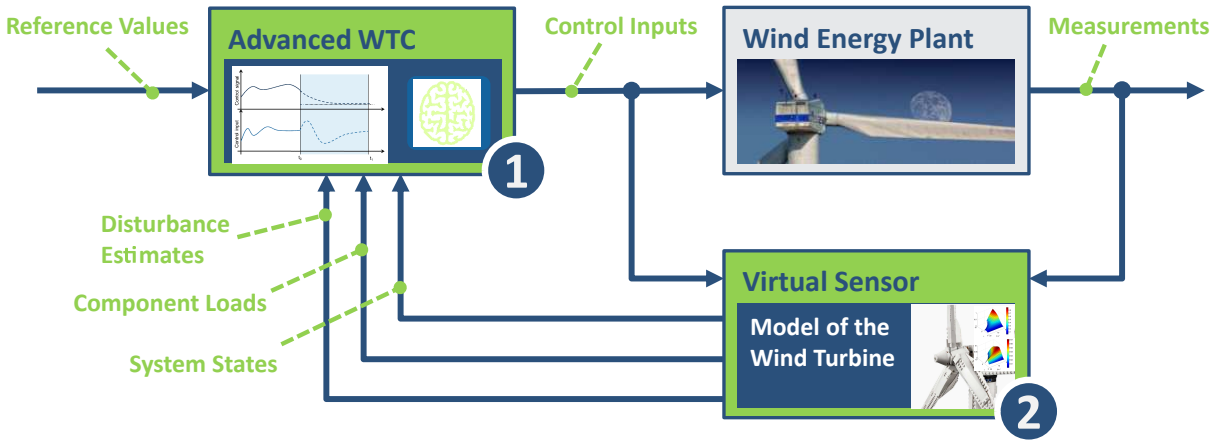

Figure 1.2: Closed-loop wind turbine control (WTC) with an advanced state-feedback controller (1) and a model-based virtual sensor (2) which is realized as augmented nonlinear state estimator

As a downside, such modern control strategies often require the precise knowledge about the current dynamic state including nacelle, drive-train and blade dynamics to outperform standard industrial controllers. The combination of MPC with nonlinear state estimation is reported for instance by Kumar et al. [112]. This knowledge is considered as vital information not only for WTC but also for condition monitoring and predictive maintenance. Hence, algorithms are needed that merge different sensor data in order to estimate hidden states, disturbances, parameters and loads (see Fig. 1.2). This is known as sensor data fusion [120] (p. 345 ff) and is the contextual focus of the present thesis.

In summary, future closed-loop WTC faces two major problems: the control problem and the estimation problem. This thesis focuses only on the second problem which must be solved in order to provide the accurate information to solve the first problem better. In this regard, the main contributions are the detailed and target-oriented investigation of state estimation techniques for horizontal-axis wind turbines, the discussion of its prerequisites and available algorithms, and the consideration of the relevant design and dynamic modeling aspects which are put into practice in a unified and interdisciplinary approach.

1.2 Literature Review

After motivating the background of this thesis in the previous section, a concise and still complete overview about the state of the art and science is given. The emphasis is put on research publications about methods to resolve relevant wind turbine estimation problems. These include foremost the wind speed and wind shear estimation, the dynamic state estimation and parameter identification. Moreover, adaptive filtering concepts including process and measurement noise estimation are considered.

1.2.1 Effective Wind Speed and Shear Estimation

In the past two decades, numerous concepts and ideas have been successfully developed and applied to obtain accurate estimates of the rotor effective wind speed (EWS). The EWS is a fictitious quantity and describes the average wind conditions over the rotor which is crucial for every large-scale wind turbine system. That is why, effective wind speed (EWS) determines essentially the current operating point of the wind turbine plant which is required for controller gain-scheduling [152, 1], for feed forward control [79, 151], for maximum power-point tracking (MPPT) and optimal control [140] as well as for operational and safety reasons [34].

Since the effective wind speed cannot be measured directly (until today) and the wind speed anemometer provides only insufficient information, various estimators for EWS have been proposed and applied to detect the unknown disturbance input. First, Ma et al. applied the Newton-Raphson's method, the extended Kalman filter (EKF) as well as the iterated EKF to estimate the EWS [123]. In the same year, Novak et al. mentioned the application of Kalman filter for rotor torque estimation for a variable-speed wind turbine [145]. Then, Bhowmik & Spée used an EWS based on MPPT to improve the closed-loop performance [16] and Sbarbaro & Peña reported the application of a nonlinear deterministic observer for that purpose [181].

In 2003 and 2004, van der Hooft & van Engelen investigated the so-called torque and power balance estimator (TBE/PBE) for EWS estimation to improve energy yield by a pseudo disturbance feed-forward control [80, 79]. A similar approach was followed by Boukhezzar & Siguerdidjane who used the Newton-Raphson's method to resolve the power balance equation and then established a state feedback controller [29].

The first overview paper was published by Ostergaard et al. [151] followed by Soltani et al. who provided a second overview on different EWS estimation concepts [198]. Therein, the power balance estimator [79], the KF/EKF-based estimator [109], the unknown input observer (UIO) [147] and the immersion and invariance method [150, 149] were discussed and compared. Odgaard & Stoustrup used the UIO in particular in connection with sensor and actuator fault detection [147]. The same year, Bottasso & Croce reported the application of an adaptive extended Kalman filter for estimation of wind states [25]. Kumar & Stol used a static EKF for the same estimation problem and tested it in simulation with a linear quadratic Gaussian (LQG) controller [111].

Bourlis & Bleijs reported the application of an adaptive Kalman filter for EWS estimation (in stall regulated variable speed wind turbines) using the covariance matching strategy to estimate the process noise online [31]. On the other hand, Mateljak et al. recommended and deployed once more the torque balance estimator successfully [126]. This estimator was also reported to be implemented on an embedded control system by Ivanović et al. [85]. In 2014, Munteanu & Besançon reported a control-based approach applying a high-gain observer for reconstructing EWS [139]. Georg et al. investigated a Takagi-Sugeno observer for EWS [50]

and later Gauterin et al. compared it to the Kalman filter algorithm [48]. The estimation of EWS using autoregressive models on lidar (light detection and ranging) data has been reported recently by Giyanani et al. [52]. Moreover, Simley & Pao propose a strategy to estimate EWS using the prospering lidar technology available as so-called lidar scanners [191].

In 2015, Jena & Rajendran provided the first complete and detailed review on estimation of effective wind speed [90]. Just recently, Khoshrodi et al. published another short review paper focussing on Kalman filter-based estimators only [106].

Apart from this, Knudsen et al. focused on simplified dynamic inflow models to exploit the a priori knowledge about the stochastic nature of the wind [109]. This approach was continued in [108] where the authors proposed the EKF for estimation of EWS and axial induction. A similar approach is reported by Henriksen et al. however the special emphasis is put on the comparison of different aerodynamic models resulting in different EKF realisations [76]. In contrast, Chen & Yu tackled the short-term wind prediction by a support vector regression based on UKF and dynamic modeling [39].

Anyhow, more recent developments focus also on the estimation of vertical and horizontal wind shear owing to their relevance for wind turbine loading and control. There are two concepts in principle to assess these wind shear components:

On the one hand, Bottasso & Riboldi and Cacciola et al. propose to estimate these shears indirectly through the turbine's dynamic response from blade load sensors [27, 37]. A detailed study on wind inflow estimation was just recently published by Bertelè et al. where the authors extracted wind shear information from load harmonics analysis [12]. Moreover, Bottasso et al. published a study on wake impingement where blade loads are used to detect the position of the wake in a wind farm [26].

On the other hand, Simley & Pao, Simley and Schlipf provide methodologies to reconstruct the shears online and ahead of time from nacelle-mounted lidar measurements [191, 192, 185]. Since lidar technology is today commercially-available and shows a promising potential, it is on its way to become state-of-the-art measurement equipment of future wind turbines. That is, different concepts like lidar, Kalman filters and internal dynamic state-space models can be realized in practice and be pooled as proposed by Towers & Jones [208].

In summary, the estimation of effective wind speed has been treated broadly in literature and is more or less established. The EWS represents the main disturbance to every wind turbine system and is at the same time its main propulsion which unfortunately cannot be affected. Nevertheless, there is still potential for research since dynamic inflow models may provide improved wind estimates and the adaptive design of wind estimators avoids too conservative static filters. Estimators for vertical and horizontal wind shear are still far less popular and developed in the research community despite the fact that these come along with several advantages.

1.2.2 Linear and Nonlinear State Estimation

State estimation for wind turbines has so far received less attention in the research community (compared to the effective wind speed estimation). Although there has been published a variety of papers, most of them treat state estimation only superficially and without in-depth study. That is why, it is often used only as a subcomponent of the wind turbine's closed-loop control circuit and the focus is most of the times laid on the controller and its design. In the following, a retrospect over the last two decades is presented in chronological order.

As one of the first, Bossanyi mentioned state estimation via Kalman filtering as a possible enabler for advanced WTC (as a side note) at the 2000 ASME Wind Energy Symposium [19]. Three years later, the author reported the successful application of a Kalman filter in a linear quadratic Gaussian (LQG) controller scheme which was used for active load reduction [21]. However, no details were provided regarding the design and performance of the estimator. The next year, Wright used a Luenberger observer to estimate the tower fore-aft and drive-train states for a linear quadratic regulator (LQR) controller [220]. Simultaneously, Stol & Fingersh reported the use of linear deterministic observers for the purpose of state-space controller testing [205]. In 2006, Wright et al. mentioned briefly the use of an observer for testing state-space controller at the Controls Advanced Research Turbines (CART) [219]. One year later, Ehlers & Diop presented the comparison of different estimators, among other a linear Kalman filter [46]. Moreover, the authors highlighted the relevance of a thorough observability analysis using the observability Gramian matrix and compared different sensor configurations. In 2008, Zhihong et al. touched briefly upon the tower fore-aft state estimation [225] though an in-depth study was still missing. The same year, Wright & Fingersh again reported the application of a deterministic observer for wind turbine state estimation [218]. Subsequently, Selvam et al. published an article on LQG control using a linear KF for individual pitch control (IPC) [190]. So far only linear or linearized Kalman filters had been applied.

However, that changed when Bottasso & Croce reported the use of an extended Kalman filter (EKF) for both state and wind estimation [24, 25]. The authors presented a new cascade architecture with multiple EKFs in parallel and verified the approach also for estimation of individual blade states. A year later, Bottasso et al. presented an update on their previous work [28]. Therein, the authors discussed again the successful application though acknowledged there might be a long way ahead to reach the necessary maturity for field testing.

Mateljak et al. reported in 2011 the use of dual estimation techniques based on two parallel EKFs [126]. Thereafter, Riboldi described in his dissertation [164] again the cascaded architecture which was introduced earlier in [24]. Subsequently, Guo & Infield proposed a (static) nonlinear state estimation technique (NSET) for condition monitoring purposes based on information from the SCADA (Supervisory Control and Data Acquisition) system [62]. In contrast, Gros et al. reported (first of all) the application of a moving horizon estimator

(MHE) as part of a state-feedback NMPC scheme [61]. The same year, Kavitha & Vijayachitra reported a single EKF for nonlinear state estimation however again no details on performance and filter design had been provided [105]. Henriksen et al. used the EKF for state estimation in connection with different aerodynamic inflow models [76]. As a result, the effects on effective wind speed estimation had been investigated.

After that, Knudsen published in 2014 a technical report on wind turbine state estimation [107]. Therein, the author reported the deployment of EKF and UKF though mentioned several problems left to be investigated, for instance observability and filter performance analysis as well as incorporation of a suitable rotor blade model. In 2015, Kumar et al. employed the unscented Kalman filter (UKF) as state estimator in a lidar-assisted MPC scheme [112].

In 2016, Jiang & Guo published a conference paper discussing again the NSET [92]. A nonlinear estimator based on EKF was presented by Vepa for rotor state estimation [211]. Finally, Sudev et al. reported in 2017 the comparison of EKF and particle filters (PF) for tower fore-aft state estimation [207].

In a nutshell, the significance of high quality state estimates is prominent for advanced wind turbine controllers and was highlighted by several of the above publications. Thus, a comprehensive treatment is essential for successful application and it is striking that this has not yet been explored in detail.

1.2.3 Model Parameter Identification

A preferably accurate model is prominent and desired in every estimation problem. That is, the model structure must be correct and, secondly, the physical and/or semi-physical parameters must be known and must match the real system's parameters. However, neither the first nor the second can be taken for granted in practice. That is why the relevant publications on identification for wind turbine application are presented hereafter.

Hansen et al. presented in 2006 new concepts to estimate the aeroelastic damping of the tower fore-aft motion [69]. In 2008, Petrović et al. touched very briefly the identification problem connected to their research on IPC [155]. Subsequently, Odgaard et al. discussed the relevance of parameter identification in order to detect changes in the aerodynamic power coefficient [146]. The concept was proposed to work without direct wind speed measurement which makes it at the same time interesting and questionable. This would mean, power curve and wind speed could be estimated independently at the same time. Simultaneously, Rose & Hiskens investigated the influence of critical system parameters on the dynamic behaviour [176]. This was done by a sensitivity analysis, followed by the definition of a nonlinear least squares problem and eventually by the solution using an iterative Gauss-Newton method.

In 2009, Iribas & Landau reported a new algorithm for closed-loop identification of the pitch control loop's discrete-time transfer function [82]. For that purpose the authors applied a

parameter adaptation algorithm (PAA) to recursively update the model parameters to the changing wind speed operating point.

In 2010, Geyler & Jasniewicz presented the outcomes on operational modal analysis (OMA) for large-scale wind turbines [51]. Therein, stochastic subspace identification (SSID) and frequency domain decomposition (FDD) were applied for identification of a linearized wind turbine model. Later that year, Jasniewicz & Geyler published further details on the identification of modal wind turbine parameters [87]. Mateljak et al. picked up on the topic and combined the parameter identification problem with the state estimation problem tackled by a dual estimation concept [126]. One year later, Petrović & Perić presented the outcomes on their study on rotor, tower fore-aft and blade model identification [156]. Unfortunately, details on the used algorithms were omitted. At the same time, Iribas-Latour & Landau published an article on closed-loop identification updating their earlier results [83]. In 2013, Devriendt et al. investigated different identification techniques for estimation of damping ratios [44]. The authors reported the particular relevance of this estimation problem for lifetime prediction.

In 2015, Weijtjens et al. presented new results on their automated OMA [214]. Therein, the authors focussed on the estimation of tower bending modes and the corresponding modal damping ratios.

Sudev et al. briefly discussed the wind turbine parameter estimation by EKF, UKF and PF [207].

1.2.4 Mechanical Load Estimation

The estimation of mechanical loads is of considerable interest for the purposes of active load reduction, condition/health monitoring and fault detection as well as design validation, among others. Generally speaking, the rotor and drive-train torques, the rotor thrust force, the blade forces and blade bending moments are considered as critical loads. There are two approaches to reconstruct them from direct or indirect measurement: first, by analysing SCADA data for condition monitoring purposes, and secondly, by load estimation using state estimation. Since the present thesis puts emphasis only on state estimation techniques, the vast number of publications on the first approach is not considered in the following.

In early 1998, Mevenkamp & Petschenka had briefly mentioned the advantages of rotor load estimation for modern wind turbine control schemes and the need for a rapid prototyping environment [135]. Henceforward, multiple publications highlighted the relevance of this estimation problem.

For instance, Hau discussed in detail the estimation of critical wind turbine component loads using Luenberger observers [73]. Therein, the relevance of accurate load models and estimator gain scheduling was highlighted. Moreover, the author gave an overview on state-of-the-art and possible future sensors in wind turbines. In the same year, Jelavić et al. reported the

estimation of blade-root and nacelle top moments with linear KF for the purpose of IPC [88]. Two years later, the authors provided an update and more details on the estimation-based IPC approach [89]. Berg & Miller presented also an approach to estimate blade loads using a KF [11] though no details on the physical model have been provided. Subsequently, Jasniewicz reported the investigation of an online estimation technique to assess mechanical loads [86]. Therein, the author presented estimation results using a deterministic observer and field test data of a modern utility-scale wind turbine.

In 2012, Bossanyi et al. published a technical report [20] where load estimation of tower bending moments was noted in a field test campaign using a Multibrid M5000 wind turbine. The authors underlined the need for filter/observer adaptation due to changing operating points and proposed to investigate in particular the estimation of critical blade loads in future.

1.2.5 Process and Measurement Noise Adaptation

Noise adaptation for wind turbine state estimation has so far received only very little attention in the wind energy research community. This is astonishing since the relevance for advanced control schemes is prevalent. There are two main reasons for noise adaptation, argued for in literature: first, the ease of tuning is somehow reduced and/or completely obsolete. Secondly, adaptive filters can tackle changing or initially incorrect assumptions on noise statistics.

In 2009 and 2010, Bottasso & Croce and Bottasso et al. employed successfully adaptive EKF schemes for wind and state estimation [25, 28]. Therein, the authors proposed a sequential adaptation approach to ease the filter tuning process and to estimate the noise covariances recursively.

Simultaneously, Bourlis & Bleijs reported the application of an adaptive filter for wind estimation [31]. Therein, the authors used the covariance matching techniques to estimate the process and measurement noise statistics. Moreover, Bourlis applied both the innovation adaptive estimation (IAE) and the multiple-model adaptive estimation (MMAE) approach to reconstruct the aerodynamic rotor torque [30].

Riboldi noted the application of adaptive EKF for state estimation which the author employed as part of a novel advanced wind turbine control scheme [164].

1.3 Accompanying Publications

In the past five years, several aspects of this thesis have been touched and presented in conference and journal papers (cf. App. C, p. 157). The first publications were intended to develop control-oriented wind turbine models and to investigate advanced control strategies and their potential for future WTC [172, 168].

From 2016 forward, the focus had been laid on state estimation (and adjacent topics) as dual problem to the controller design. Preliminary results on state estimation using sigma-point filters were addressed in [167]. Besides the often used extended Kalman filter (EKF), the unscented Kalman filter (UKF) and central difference Kalman filter (CDKF) were proposed as more robust algorithms for joint state and effective wind speed estimation.

Subsequently, the complete estimation problem was addressed at the TORQUE conference in Munich [174]. The relevant five sub-problems were formulated, namely the estimation of disturbance inputs, the estimation of hidden dynamic states, the estimation of process and measurement noise covariances, the online identification of model parameters and the estimation of critical component loads. For that purpose, a tailored observer architecture for wind turbines was proposed based on sigma-point Kalman filters (SPKF) and linear steady-state Kalman filters. In the aftermath, it became clear that practical considerations regarding the algorithm and the filter design needed a further and more thorough investigation.

Hence, practical implementation issues like real-time capabilities and observability were discussed in [170]. Therein, the square-root cubature Kalman filter (SR-CKF) was used as an advantageous algorithm in order to address the estimation problem of blade dynamic states and blade-root bending moments. This problem was explored as an illustrative and relevant sub-problem for different sensor configurations to evaluate the expectable estimation accuracy, comparing state-of-the-art and future sensors.

As a matter of fact, noise adaptation was highlighted in [174] as one important aspect in the agenda of making nonlinear state estimation ready for wind turbine application. Yet, this topic had not been addressed thoroughly for wind turbine application. For this reason, two follow-up contributions aimed specifically at the filter adaptation in particular, discussing simulation results using adaptive sigma-point filters [173, 166]. Therein, preliminary results for the design and the performance assessment of master-slave filters and a maximum likelihood estimator were explored. Hence, special attention was given to the adaptation of the filter parameters and the estimation of noise parameters, highlighting the advantageous aspects in an unknown and time-varying noise environment.

In previous publications, the relevance of a detailed observability study had been mentioned often but only scratch the surface (cf. [170]). For this reason, a more detailed research focusing on observability analysis for wind turbines only was presented in late 2018 [169]. Therein, empirical Gramian matrices were used to derive and analyse criteria for observability. Special attention was directed at the expectable estimation accuracy with the different sensor configurations. Both, the observability criteria and the performance, were determined as the two sides of the same coin needed to be addressed to evaluate the practical observability.

Finally, the latest publication on adaptive filtering [171] further develops the classical CKF in order to cope with time-varying noise statistics as well as inequality constraints on the estimated states. The resulting adaptive filter is suggested to provide more accurate state estimates and to be more robust against filter divergence. Moreover, the contribution pro-

poses an automated filter design based on numerical optimization which uses the normalized estimation error squared (NEES) and the normalized innovation squared (NIS) as part of the objective function. The novel adaptive CKF is applied to wind turbines in order to assess the potential improvement for state and parameter estimation. The simulation results for an illustrative acid test scenario with time-varying measurement noise show the superiority of the novel adaptive CKF since it compensates the noise robustly and thereby outperforms the classical filter.

1.4 Research Objectives

This thesis focuses on the generic state estimation problem for horizontal axis wind turbines (HAWT). These turbines are currently the predominant type and also the most promising concept for future wind energy conversion on a large scale. Therefore, a complete treatment on state estimation for modern HAWT is indispensable. As highlighted in Sect. 1.2, many aspects of wind turbine state and parameter estimation have been touched by other authors before. Though, no comprehensive study for wind turbine application is available today. As preparatory work for the present work, several publications have addressed accompanying topics and issues to pave the way (cf. Sect. 1.3). The present thesis shall bridge the existing gap between the estimation theory on the one hand and the wind turbine engineering practise on the other hand.

The main goal is to conduct a profound investigation of nonlinear state estimation techniques for wind turbine application. Thus, a closer look at the critical aspects is needed to promote state estimation for the industry-oriented application. For instance, the eligible algorithms must be selected and evaluated. The target-oriented physical modeling must be specified and conducted. The methodology for the free design parameters of the filter must be developed and tested. The designed estimator must be tested and assessed in a thorough simulation study. Finally, a dedicated tool chain needs to be developed in order to reach all the previously mentioned aspects in an automated manner.

There are two key points to be addressed in this research project. Obviously, both are intertwined to a certain degree:

- The first point is the definition of **the estimation problem** including the available sensor data, the system model, the desired quantities to be estimated and further practical constraints. The main estimation problem constitutes in short:

  > Estimate the nacelle dynamics, the drive-train dynamics and the effective wind speed without direct measurement of the states and disturbances and without the knowledge of the underlying process and measurement noise dynamics, only using the already available sensor information!

⇒ This is considered as the application-specific perspective, answering the questions: *What is the purpose and the application of the estimator? And what are the practical requirements?*

- The second point is the definition and solution of **the design problem** including amongst others the design parameters of the estimator, the design methodology as well as the performance indicators and objective function, respectively. The main design problem constitutes:

  > Find the best estimator design for the above estimation problem that establishes the desired estimator performance for all relevant test scenario cases and show the limitations of the design methodology!

  ⇒ This is considered as the mathematical perspective, answering the question: *How is the best estimator design found?*

In conclusion, the research questions of the present thesis, derived from the above explanations, are identified as follows:

- What are the requirements for successful application of nonlinear state estimation to wind turbines?
- Which estimators and algorithms suit the practical needs best?
- What are the benefits and limitations of supplementary components for adaptation? When is it useful to estimate noise covariances online?
- Which control-oriented models are needed as filter models? What is the required model granularity?
- Which approach is favourable to find the best set of filter parameters? How to formulate the design problem and how to solve it?
- What are the suitable measures to assess the estimator's performance? Which information does the estimator provide itself for that purpose?
- How do the designed estimators perform in a realistic simulation environment under adverse conditions?
- What are the limitations of the proposed approach?

By systematically answering of the above questions, the dissertation shall contribute to the body of wind energy research. It shall broaden in addition the knowledge about dynamic modeling and highlight the advantages of state estimation for wind turbine application.

1.5 Dissertation Outline

This section is intended to give the valued reader a concise overview on the dissertation's contents. The outline reads as follows:

Chapter 2 provides a compressed introduction to the Kalman filter (KF) algorithms which are used for state estimation and sensor data fusion tasks, respectively. First, the linear KF is described very briefly to provide just the necessary facts and notations. Subsequently, the sigma-point approach for nonlinear control systems is explained by introducing the classical cubature Kalman filter as well as its square-root version. This is followed by the definition of the relevant criteria to assess the estimation performance. Some of these performance indicators are also needed for the filter adaptation. In particular, the master-slave approach and the Maybeck estimator are presented as two used adaptive filters. The chapter brings together and extends the relevant aspects to make the thesis self-consistent and gives the reader a concise initiation.

The following **Chapter 3** is devoted to the control-oriented physical white-box modeling of wind turbines. Therefore, the physical basics on wind energy conversion are introduced first and a brief summary on wind turbine technology is given for the control engineer. This includes the description of structure, components and actuators as well as the fundamentals of wind turbine control. Hereafter, different wind turbine models are introduced which serve as internal models for the model-based filter algorithms and as simulation models. The conducted white-box modeling approach incorporates only the turbine dynamics that are directly relevant for power production and fatigue load control (speaking of the drive-train, the nacelle dynamics and tower base loads).

Chapter 4 deals with the optimal design methodology for the classical and adaptive SPKF in order to replace the empirical trial-and-error-based design approach. For this purpose, a generic estimation problem is defined first that is in each case adapted to the specific problem and configuration. As a matter of fact, the problem is then solved numerically to achieve the desired design which is considered as optimum. First, this methodology is applied to find the optimal static filter design which is evaluated with different scenarios. This is followed by the filter design for an adaptive linear slave filter. The chapter closes with a summary and discussion on the novel design approach.

Chapter 5 deals with the in-depth simulation study of the designed estimators with and without filter parameter adaptation. For this purpose, the developed engineering suite with its capabilities for the automatic filter performance assessment (FPA for short) is introduced first. Hereafter, the test scenarios are defined in order to allow for a preferably realistic simulation study of the developed estimators. These scenarios are evaluated with the FPA tool and various filter designs are discussed. The chapter also includes additional results for advanced problems, obtained from preparatory studies, to show the wide-range application possibilities of state estimators. For instance, the comparison of a distributed observer architecture and the technology transfer to tidal turbines is illustrated.

Finally, **Chapter 6** summarizes the main outcomes of the thesis and discusses the novel results in a concise manner. The achievements are put into a greater context and the relevance of state estimation for advanced control schemes is highlighted once more. Last but not least, the thesis concludes with recommendations for the next practical steps and thus also opens pathways for future research in the field of wind turbine state estimation.

2 Kalman Filtering

This chapter presents the necessary background information on Kalman filtering for linear and nonlinear control systems. It introduces the relevant performance criteria which are later used for filter performance assessment and also serve as measurement for filter parameter adaptation. In addition, two different cubature Kalman filters are presented and two adaptation rules are proposed to cope with unknown or uncertain noise environment.

2.1 Introduction

An estimator is an auxiliary static or dynamic system [13] that runs often in parallel to the real system under investigation (online) in order to estimate desired system quantities. Depending on the nature of the algorithm, a stochastic estimator is denoted as a filter whereas a deterministic estimator is widely known as an observer. Generally speaking, the main prerequisite for a successful estimation using an estimator (algorithm) constitutes that the system properties of observability and identifiability, respectively, are granted.

In 1960, Rudolf E. Kalman published his seminal paper introducing a new algorithm for prediction and filtering [104]. This nowadays-called Kalman filter (KF) is an optimal filter (or more specifically an optimal stochastic estimator) which is well suited to obtain the desired hidden quantities from noisy measurements. These quantities are often unmeasurable, uncertain and/or hidden information of a real-world system which are often dynamic states or model parameters. The main advantages of the KF are the recursive evaluation and the handling of non-stationary processes by nature, for instance compared to the optimal Wiener filter [217] or the least-squares estimator, and also the handling of dynamic systems. Hence, the KF facilitates a low computational effort as well as low memory requirements which have been decisive back in the days. Additionally, it offers a suitable alternative to compute new filter estimates as soon as new measurement information arrive. Sorenson [202] describes the historical development starting from Gauss' ideas to the invention of the KF. More details about the algorithms for linear time-(in)variant systems are provided in Sect. 2.2 (p. 17 ff).

Since many technical systems cannot be described accurately enough by linear state-space models, nonlinear extensions of the linear KF have been developed to allow for nonlinear state estimation based on Kalman's algorithm. Nonlinear stochastic estimators are distinguished between so-called local and global filters. The first rely on certain approximations like local linearization and/or the assumption of Gaussianity for the random variables (RV).

On the contrary, global filters allow for arbitrary probability density functions (pdf) of the RV. A detailed treatment of these global filters is excluded for sake of brevity and because derivative-less local filters are found to be well sufficient in the context of this research project. A detailed overview is given for instance by Šimandl [212]. A brief discussion of local and global filters for wind turbine application provides [174].

The typical representative of a standard local filter is the extended Kalman filter (EKF) [188, 102, 216, 194]. It is still popular and widely used for various industrial applications due to its simplicity (in particular when the derivatives are available analytically). The EKF incorporates the system's nonlinearity often effectively by a recursive linearization procedure around the previous state estimate. However, due to this feedback character the risk of filter instability rises, cf. [134, 132] and [120] (p. 347). Additionally, all standard local filters use approximations only, even if the nonlinear model is accurately known. This turns out to be an unattractive approach since it is avoidable nowadays.

It was not before 1995 when Uhlmann discovered that *"it is easier to approximate a Gaussian distribution than it is to approximate an arbitrary nonlinear function or transformation"* [209]. This insight had led finally to the invention of the so-called unscented transform[1] and a new extension of the Kalman filter to nonlinear control systems which is argued to be more accurate and easier to implement than the EKF [101].

Fortunately, this new exploration means that the standard local filters are actually obsolete today because of the availability of the so-called derivative-less alternatives. These newer types of (derivative-less) local filters eliminate the necessity to evaluate partial derivatives of the system equations completely [101, 180]. Instead, they employ so-called sigma-points which are deterministically chosen representative vector points for the assumed Gaussian multivariate density distributions. These filters are summarized in the class of sigma-point Kalman filters (SPKF) [132, 74]. The first member of this family was the unscented Kalman filter (UKF) [101, 213]. Further details on the selected SPKF algorithms are provided in Sect. 2.3 (p. 21 ff).

Although the standard SPKF tackles the system's nonlinearity effectively, it cannot manage the problem of changing and/or unknown noise statistics by nature. Lack of knowledge about these noise statistics is often present in many technical systems. Furthermore, an unfortunate design may lead to filter divergence or severe performance degradation in practice. One way to deal with these issues is to apply a very conservative filter design which though may well cause poor estimation performance itself. A more elegant approach is to incorporate an automated adjustment feature into the algorithm which is denoted as filter adaptation. This strategy enables the estimator to change incorrect filter parameters online, for instance elements of the noise covariance matrices. Further details on the adaptation rules (investigated and realized in this thesis) are found in Sect. 2.5 (p. 33 ff).

[1] Please refer to an interview with Jeffrey Uhlmann for in-depth and first-hand explanations on his thoughts with respect to the invention of the unscented transform and the unscented Kalman filter `https://ethw.org/First-Hand:The_Unscented_Transform` [accessed 2020-01-16]

The required performance measures, which are used for filter performance assessment and filter parameter adaptation, are introduced in Sect. 2.4 (p. 27 ff).

Finally, the concept of observability is relevant when talking about state estimation and the concept of the Kalman filter, respectively. It is not by accident that both concepts had been developed in parallel and also by the similar persons, for instance by Rudolph E. Kalman. The reason is that the best estimator and estimator design cannot cope with an unobservable system. Such a system basically is not suited for application in feedback-controlled systems since the measurements do not contain enough information to guarantee the expected estimation accuracy. Hence, one mandatory step prior to application or field testing is the thorough investigation of observability. Thereby, possible filter divergence (due to non-observability) shall be assuredly excluded in order to facilitate the practical application to succeed. This analysis was done in a pre-study and published in [169].

Though, it must be noted that an unobservable system can still be useful in terms of forward simulation. This is eligible when the system is stable but partially or fully unobservable (hence a detectable system). Therewith, the filter produces additional estimates that are not mission-critical, though are used for instance to monitor mechanical loads on the tower or on the blades.

2.2 Linear Kalman Filters

The linear Kalman filter has been properly discussed in multiple publications and books over the last decades [104, 103, 32, 38, 58, 193, 194, 41, 57]. Hence in the following, only a concise introduction is provided in order to explain the used notation of the thesis and to lay the foundation to understand the adaptation concepts later on.

One fundamental prerequisite for applying the standard KF is a linear (discrete-time) state-space representation of the investigated system defined by

$$\boldsymbol{x}_{k+1} = \mathbf{A}_k \boldsymbol{x}_k + \mathbf{B}_k \boldsymbol{u}_k + \mathbf{G}_k \boldsymbol{w}_k \tag{2.1a}$$

$$\boldsymbol{y}_k = \mathbf{C}_k \boldsymbol{x}_k + \mathbf{D}_k \boldsymbol{u}_k + \boldsymbol{n}_k \tag{2.1b}$$

where the following vector quantities are used: the state $\boldsymbol{x}_k$, the control input $\boldsymbol{u}_k$, the output or measurement $\boldsymbol{y}_k$, the process noise $\boldsymbol{w}_k$ and the measurement noise $\boldsymbol{n}_k$. The system matrices $\mathbf{A}_k$, $\mathbf{B}_k$, $\mathbf{C}_k$, $\mathbf{D}_k$ and $\mathbf{G}_k$ are assumed to be a priori known and are possibly time-variant quantities. The index k represents the considered sample point in time $t_k = k\,T_\mathrm{s}$ where T_s denotes the sample time.[2]

A special case of Eq. (2.1) is the linear time-invariant system

$$\boldsymbol{x}_{k+1} = \mathbf{A}_\mathrm{d} \boldsymbol{x}_k + \mathbf{B}_\mathrm{d} \boldsymbol{u}_k + \mathbf{G}_\mathrm{d} \boldsymbol{w}_k \tag{2.2a}$$

$$\boldsymbol{y}_k = \mathbf{C}_\mathrm{d} \boldsymbol{x}_k + \mathbf{D}_\mathrm{d} \boldsymbol{u}_k + \boldsymbol{n}_k \tag{2.2b}$$

[2] For instance, $\boldsymbol{x}_k$ is the abbreviated notation of the state vector $\boldsymbol{x}(t_k)$

with constant system matrices $\mathbf{A}_\mathrm{d}$, $\mathbf{B}_\mathrm{d}$, $\mathbf{C}_\mathrm{d}$, $\mathbf{D}_\mathrm{d}$ and $\mathbf{G}_\mathrm{d}$.

The concept of a dynamic state is fundamental to the above state-space representation. The state contains the (ideally minimum) amount of information necessary to determine the current *situation* of a system uniquely. The number of observations n_y (dimension of $\boldsymbol{y}_k$) is most of the times smaller than the state vector's dimension n_x. That is why, Eq. (2.1b) often cannot be inverted directly in order to obtain $\boldsymbol{x}_k$ because $\mathbf{C}_k^{-1}$ does not exist. The KF circumvents this problem elegantly and infers the unknown/desired initial state vector $\boldsymbol{x}_0$ from the known inputs $\boldsymbol{u}_k$ and the observations $\boldsymbol{y}_k$ by recursive evaluation over time (without a need to invert the measurement equation).

The Kalman filter is well-known to be the optimal filter for linear systems [194]. Though, there are a few underlying assumptions to fulfil the optimality criterion:

- First, the linearity of the system (2.1) is required and, moreover, the system matrices must be accurately known prior to application.
- Secondly, the process noise $\boldsymbol{w}_k$ and the measurement noise $\boldsymbol{n}_k$ have to be Gaussian, uncorrelated, zero-mean and white (and also must be known in advance).
- The third condition pertains to the system's property of observability. Accordingly, the system matrix $\mathbf{A}_k$ and the output matrix $\mathbf{C}_k$ must fulfil certain properties in order to enable the Kalman filter to estimate the state vector $\boldsymbol{x}_k$. For instance, Kalman's observability criterion [197] is useful in this context.

Mathematically, the second item is described by normal distributed noise

$$\boldsymbol{q}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_k) \quad \text{and} \quad \boldsymbol{r}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_k) \tag{2.3}$$

with the process and measurement noise covariances $\mathbf{Q}_k$ and $\mathbf{R}_k$. The whiteness and uncorrelatedness are presupposed with

$$E\Big\{\boldsymbol{w}_k \boldsymbol{w}_l^\mathrm{T}\Big\} = \mathbf{Q}_k \delta_{k-l} \ , \quad E\Big\{\boldsymbol{n}_k \boldsymbol{n}_l^\mathrm{T}\Big\} = \mathbf{R}_k \delta_{k-l} \ , \quad E\Big\{\boldsymbol{w}_k \boldsymbol{n}_l^\mathrm{T}\Big\} = \mathbf{0} \tag{2.4}$$

using the expectation operator $E\{\cdot\}$ and Kronecker's delta function δ_{k-l}. The covariance matrices $\mathbf{Q}_k$ and $\mathbf{R}_k$ must be also known a priori.

On the above conditions, the Kalman filter obtains the most likely state estimate $\hat{\boldsymbol{x}}_k$ which is unbiased and has the minimum error variance of all possible filters [58]. Particularly, all linear and nonlinear discrete-time KF algorithms provide at least two estimates for the same state: one a priori (predicted) estimate $\hat{\boldsymbol{x}}_k^-$ with covariance $\mathbf{P}_k^-$ defined by the conditional expectations

$$\hat{\boldsymbol{x}}_k^- = E\Big\{\boldsymbol{x}_k | \boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_{k-1}\Big\} \tag{2.5a}$$

$$\mathbf{P}_k^- = E\Big\{(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k^-)(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k^-)^\mathrm{T}\Big\} \tag{2.5b}$$

which is known as prediction step or time-update step. The second estimate is called posterior (corrected) estimate $\hat{\boldsymbol{x}}_k^+$ with covariance $\mathbf{P}_k^+$ defined by

$$\hat{\boldsymbol{x}}_k^+ = E\{\boldsymbol{x}_k|\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_k\} \tag{2.6a}$$

$$\mathbf{P}_k^+ = E\Big\{(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k^+)(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k^+)^{\mathrm{T}}\Big\} \tag{2.6b}$$

which is denoted correction step or measurement-update step. Both estimates are related by the KF's update equations

$$\hat{\boldsymbol{x}}_k^+ = \hat{\boldsymbol{x}}_k^- + \boldsymbol{\mathcal{K}}_k \boldsymbol{v}_k \tag{2.7a}$$

$$\mathbf{P}_k^+ = \mathbf{P}_k^- - \boldsymbol{\mathcal{K}}_k \mathbf{P}_k^{yy} \boldsymbol{\mathcal{K}}_k^{\mathrm{T}} \tag{2.7b}$$

where $\boldsymbol{\mathcal{K}}_k$ represents the Kalman gain matrix and $\boldsymbol{v}_k$ the filter's innovation vector. $\mathbf{P}_k^{yy}$ is innovation covariance matrix defined by

$$\mathbf{P}_k^{yy} = E\Big\{(\boldsymbol{y}_k - \hat{\boldsymbol{y}}_k^-)(\boldsymbol{y}_k - \hat{\boldsymbol{y}}_k^-)^{\mathrm{T}}\Big\} = E\Big\{\boldsymbol{v}_k \boldsymbol{v}_k^{\mathrm{T}}\Big\} \quad . \tag{2.8}$$

The Kalman gain $\boldsymbol{\mathcal{K}}_k$ represents loosely speaking a confidence measure whether to trust the prediction $\hat{\boldsymbol{x}}_k^-$ or the innovation sequence $\boldsymbol{v}_k$. The latter is the new information to the filter from current observation $\boldsymbol{y}_k$ of time step k that is given by

$$\boldsymbol{v}_k = \boldsymbol{y}_k - \hat{\boldsymbol{y}}_k \,. \tag{2.9}$$

Rearranging Eq. (2.7a) then yields

$$\hat{\boldsymbol{x}}_k^+ - \hat{\boldsymbol{x}}_k^- = \boldsymbol{\mathcal{K}}_k \boldsymbol{v}_k = \widetilde{\boldsymbol{x}}_k \tag{2.10}$$

where $\widetilde{\boldsymbol{x}}_k$ is denoted as residual of the state estimate (state residual) which is equally important for filter performance evaluation (see Sect. 2.4) and the adaptive KF variants (Sect. 2.5).

The necessary background to evaluate the expectations in Eqs. (2.5), (2.6) and (2.8) with a view to derive, among others, the Kalman gain is comprehensively treated in [194, 59, 215, 127]. Therefore, only the relevant equations are put together in Algorithm 2.1. These are applicable for linear time-variant (LTV) systems like (2.1) only.

As alternative for Eq. (2.7b) and (2.11i), the so-called Joseph's form

$$\mathbf{P}_k^+ = \Big(\mathbf{I} - \boldsymbol{\mathcal{K}}_k \mathbf{C}_k\Big)\mathbf{P}_k^-\Big(\mathbf{I} - \boldsymbol{\mathcal{K}}_k \mathbf{C}_k\Big)^{\mathrm{T}} + \boldsymbol{\mathcal{K}}_k \mathbf{R}_k \boldsymbol{\mathcal{K}}_k^{\mathrm{T}} \tag{2.13}$$

is recommended for higher robustness to ill-conditioned problems [194] due to greater numerical accuracy which though increases the computational effort [18, 224].

A special case of the discrete-time KF constitutes the steady-state Kalman filter [41]. The Algorithm 2.2 presents the main equations which are applicable for linear time-invariant (LTI) systems only. Therein, the steady-state Kalman gain matrix $\boldsymbol{\mathcal{K}}_{\mathcal{S}}$ and the steady-state covariance matrices $\mathbf{P}_{\mathcal{S}}^-$, $\mathbf{P}_{\mathcal{S}}^+$ and $\mathbf{P}_{\mathcal{S}}^{xy}$ are constant and evaluated by the solution of discrete-time algebraic Riccati equation (DARE) as the initialization step. This comes along with

Algorithm 2.1 (Standard Kalman filter for linear time-variant systems)

Initialization step:

$$\text{Initial State Mean} \qquad \hat{\boldsymbol{x}}_0^+ = E\{\boldsymbol{x}_0\} \tag{2.11a}$$

$$\text{Initial State Covariance} \qquad \mathbf{P}_0^+ = E\Big\{(\boldsymbol{x}_0 - \hat{\boldsymbol{x}}_0)(\boldsymbol{x}_0 - \hat{\boldsymbol{x}}_0)^{\mathrm{T}}\Big\} \tag{2.11b}$$

Prediction step:

$$\text{Predicted state} \qquad \hat{\boldsymbol{x}}_k^- = \mathbf{A}_{k-1}\hat{\boldsymbol{x}}_{k-1}^+ + \mathbf{B}_{k-1}\boldsymbol{u}_{k-1} \tag{2.11c}$$

$$\text{Predicted error covariance} \qquad \mathbf{P}_k^- = \mathbf{A}_{k-1}\mathbf{P}_{k-1}^+\mathbf{A}_{k-1}^{\mathrm{T}} + \mathbf{G}_{k-1}\mathbf{Q}_k\mathbf{G}_{k-1}^{\mathrm{T}} \tag{2.11d}$$

$$\text{Predicted output} \qquad \hat{\boldsymbol{y}}_k^- = \mathbf{C}_k\hat{\boldsymbol{x}}_k^- + \mathbf{D}_k\boldsymbol{u}_k \tag{2.11e}$$

Correction step:

$$\text{Cross Covariance Matrix} \qquad \mathbf{P}_k^{xy} = \mathbf{P}_k^-\mathbf{C}_k^{\mathrm{T}} \tag{2.11f}$$

$$\text{Innovation Covariance Matrix} \qquad \mathbf{P}_k^{yy} = \mathbf{C}_k\mathbf{P}_k^-\mathbf{C}_k^{\mathrm{T}} + \mathbf{R}_k \tag{2.11g}$$

$$\text{Kalman Gain} \qquad \boldsymbol{\mathcal{K}}_k = \mathbf{P}_k^{xy}(\mathbf{P}_k^{yy})^{-1} \tag{2.11h}$$

$$\text{Corrected Error Covariance} \qquad \mathbf{P}_k^+ = \mathbf{P}_k^- - \boldsymbol{\mathcal{K}}_k\mathbf{P}_k^{yy}\boldsymbol{\mathcal{K}}_k^{\mathrm{T}} \tag{2.11i}$$

$$\text{Filter Innovation} \qquad \boldsymbol{v}_k = \boldsymbol{y}_k - \hat{\boldsymbol{y}}_k^- \tag{2.11j}$$

$$\text{Corrected state estimate} \qquad \hat{\boldsymbol{x}}_k^+ = \hat{\boldsymbol{x}}_k^- + \boldsymbol{\mathcal{K}}_k\boldsymbol{v}_k \tag{2.11k}$$

Algorithm 2.2 (Steady-state Kalman filter for linear time-invariant systems)

Initialization step:

$$\text{Initial State Mean} \qquad \hat{\boldsymbol{x}}_0^+ = E\{\boldsymbol{x}_0\} \tag{2.12a}$$

$$\text{Initial State Covariance} \qquad \mathbf{P}_0^+ = E\Big\{(\boldsymbol{x}_0 - \hat{\boldsymbol{x}}_0)(\boldsymbol{x}_0 - \hat{\boldsymbol{x}}_0)^{\mathrm{T}}\Big\} \tag{2.12b}$$

$$\text{Steady-state solution of DARE} \qquad \mathbf{P}_{\mathcal{S}}^- \,,\ \mathbf{P}_{\mathcal{S}}^+ \tag{2.12c}$$

$$\text{Steady-state cross covariance} \qquad \mathbf{P}_{\mathcal{S}}^{xy} = \mathbf{P}_{\mathcal{S}}^-\mathbf{C}_{\mathrm{d}}^{\mathrm{T}} \tag{2.12d}$$

$$\text{Steady-state innovation covariance} \qquad \mathbf{P}_{\mathcal{S}}^{yy} = \mathbf{C}_{\mathrm{d}}\mathbf{P}_{\mathcal{S}}^-\mathbf{C}_{\mathrm{d}}^{\mathrm{T}} + \mathbf{R}_{\mathrm{d}} \tag{2.12e}$$

$$\text{Steady-state Kalman Gain} \qquad \boldsymbol{\mathcal{K}}_{\mathcal{S}} = \mathbf{P}_{\mathcal{S}}^{xy}(\mathbf{P}_{\mathcal{S}}^{yy})^{-1} \tag{2.12f}$$

$$\text{or} \qquad \boldsymbol{\mathcal{K}}_{\mathcal{S}} = \mathbf{P}_{\mathcal{S}}^+\mathbf{C}_{\mathrm{d}}^{\mathrm{T}}\mathbf{R}_{\mathrm{d}}^{-1} \tag{2.12g}$$

Prediction step:

$$\text{Predicted state} \qquad \hat{\boldsymbol{x}}_k^- = \mathbf{A}_{\mathrm{d}}\hat{\boldsymbol{x}}_{k-1}^+ + \mathbf{B}_{\mathrm{d}}\boldsymbol{u}_{k-1} \tag{2.12h}$$

$$\text{Predicted output} \qquad \hat{\boldsymbol{y}}_k^- = \mathbf{C}_{\mathrm{d}}\hat{\boldsymbol{x}}_k^- + \mathbf{D}_{\mathrm{d}}\boldsymbol{u}_k \tag{2.12i}$$

Correction step:

$$\text{Filter Innovation} \qquad \boldsymbol{v}_k = \boldsymbol{y}_k - \hat{\boldsymbol{y}}_k^- \tag{2.12j}$$

$$\text{Corrected state estimate} \qquad \hat{\boldsymbol{x}}_k^+ = \hat{\boldsymbol{x}}_k^- + \boldsymbol{\mathcal{K}}_{\mathcal{S}}\boldsymbol{v}_k \tag{2.12k}$$

a reduced computational effort (during recursive execution of the prediction and correction step) since the covariance matrices and the Kalman gain are not computed in each step any more. The benefits from that are a reduced number of matrix computations, the avoidance of matrix inversions and, hence, less divergence issues are likely to occur. The steady-state Kalman filter is applicable, for instance, as state estimator for the mechanical wind turbine states in a distributed observer architecture (cf. Sect. 5.5.1, p. 123).

2.3 Sigma-Point Kalman Filters

In hindsight, the development of the SPKF was originally initiated by Uhlmann [209] in 1995. After that, his findings had led first to the invention of the UKF in 1996/97, introducing a new approach to Kalman filtering [99, 101], based on the so-called unscented transform (UT). Finally, these new types of derivative-less filters were generalized to a joint framework which is today referred to as the SPKF class [132, 74]. Generally speaking, the SPKF is advantageous whenever physical and non-physical systems cannot be described accurately by local linear models and thus nonlinear representations must be used.

2.3.1 A General Overview

The term SPKF refers to filter algorithms that work with so-called sigma points (SP), or sometimes called cubature points. These are deterministically chosen vector points in the state space with associated weighting factors. This set of points is transformed easier through a nonlinear function than an entire probability density function of the original random variable [101]. In these days, the SPKF class includes the following members:[3]

- The unscented Kalman filter (UKF) [99, 101, 194],
- the spherical simplex Kalman filter (SSKF) [100, 74],
- the central/finite difference Kalman filter (CDKF/FDKF) [182, 132, 74],
- the divided difference Kalman filter (DDKF) [144, 143],
- the Gauss-Hermite Kalman filter (GHKF) [5, 179, 74, 180], and
- the cubature Kalman filter (CKF) [7, 6, 180].

The above family members differ in the number of evaluated SP as well as in the choice of the numerical integration rule to approximate the multivariate probability density of the n_x random variables (hence the origin of the weights).

[3] Today, there are also numerically more robust square-root versions available [133, 134, 132, 4, 7, 71]. These algorithms propagate directly the Cholesky factor (matrix square-root) of the estimation error covariance matrix, improving the filter performance for instance with round-off errors or too clean measurements.

For instance, the UKF uses the UT to determine the weights and the $2n_x + 1$ SP. So does the SSKF, though the spherical simplex approach provides a numerically better-behaved selection strategy for the UT using $n_x + 1$ SP only [100]. The CDKF is very similar to the UKF, yet interestingly also related to the second-order EKF (SOEKF) since it approximates the nonlinear model by a second-order Stirling polynomial interpolation. Instead of simple polynomials, the GHKF uses orthogonal Hermite polynomials to approximate the multi-dimensional distribution [74]. Due to an exponentially increasing number of SP, the GHKF suffers from the curse of dimensionality [43] (cf. Sect. 2.3.2). In contrast, the CKF uses a cubature integration rule which makes it well-suited for higher-order systems as well. It is supposed to be even superior to the UKF [7]. The CKF requires only $2n_x$ SP to support the statistic properties. Hence, the computational costs are similar compared to UKF/CDKF.

In summary, all SPKF have in common that they do not linearize the model but rather than linearize the noise statistics since these are approximated by a statistical linear regression approach. That is why, SPKF are sometimes also denoted as linear regression Kalman filters (LRKF) [117].

Generally speaking, the SPKF algorithm is always then valuable whenever linear state-space models (2.1) are not any longer sufficient. Then, nonlinear state-space models like

$$\boldsymbol{x}_{k+1} = \boldsymbol{f}(\boldsymbol{x}_k, \boldsymbol{u}_k) + \mathbf{G}_k \boldsymbol{w}_k \tag{2.14a}$$

$$\boldsymbol{y}_k = \boldsymbol{h}(\boldsymbol{x}_k, \boldsymbol{u}_k) + \boldsymbol{n}_k \tag{2.14b}$$

come into play in order to characterize the system dynamics over a greater operating range appropriately. In this case, the assumption of additive process noise $\boldsymbol{w}_k$ and additive measurement noise $\boldsymbol{n}_k$ is implied.[4] As discussed briefly in Sect. 2.1 (p. 16), the SPKF exhibits practical advantages compared to the well-known EKF which needs to evaluate at least the first-order partial derivatives (Jacobians)

$$\mathbf{A}_k = \left.\frac{\partial \boldsymbol{f}(\cdot)}{\partial \boldsymbol{x}_k}\right|_{\hat{\boldsymbol{x}}_{k-1}^{+}} \quad \text{and} \quad \mathbf{C}_k = \left.\frac{\partial \boldsymbol{h}(\cdot)}{\partial \boldsymbol{x}_k}\right|_{\hat{\boldsymbol{x}}_{k}^{-}} \tag{2.15}$$

in every recursion step or possibly also the second-order partial derivatives (Hessians) for the SOEKF [63]. In accordance with [99, 101, 194], the paramount advantages, which advocate for using the SPKF in practice, are:

- Accuracy: The statistical properties (mean and covariances) of the random variables are approximated more accurately and the filter runs more robustly.
- Library usage: Due to the lack of partial derivative, SPKF work well in filter libraries since directly usable independent of the model class. This is in particular true for applications with black-box models.

[4] Refer to the publications [134, 132, 223] regarding UKF/SPKF extensions for the non-additive case whenever the additive noise assumption does not hold.

- Ease of implementation: The algorithm is comparatively easy to implement and less error-prone. Especially, if the Jacobians (and the Hessians) are not available analytically, but must be assessed in every recursion numerically.
- Enhanced class of models: The SPKF avoids local linear model approximations completely which makes them also directly applicable for non-differentiable or discontinuous functions (if and when model derivatives do not exist).

For these reasons, only SPKF are considered in the further course of this thesis. The remainder of this section introduces the CKF and its square-root version (SR-CKF) which have been chosen as representative SPKF types.

In addition, several other types have been implemented and tested in preliminary studies as well, though are not further considered here for the sake of brevity. Please refer to App. A for a brief summary of more standard and square-root algorithms and/or consult the pertinent literature for more details (see p. 21).

2.3.2 Standard Cubature Kalman Filter

The CKF was first proposed in 2009 by Arasaratnam & Haykin as a sub-optimal solution for nonlinear Bayesian estimation problems [7]. The underlying idea of this local nonlinear filter is to find an approximate solution that matches closely certain statistical moments while still being solvable in real-time.

In general, there are product rule and non-product rule based approaches to be distinguished that can be applied to approximate the multi-dimensional integrals. For instance, the aforementioned GHKF uses a quadrature rule (product rule) based on Gauss-Hermite polynomials. This filter exhibits a decent accuracy, though it comes along with m^{n_x} cubature points as samples (where m denotes the number of points per dimension, typically $m=3$). Due to this exponential law, the GHKF suffers from the curse of dimensionality which makes it no reasonable choice for $n_x > 5$ [5].

Contrary to GHKF, the CKF uses a spherical-radial cubature rule (non-product rule) that requires in total only $2n_x$ sigma-points to propagate the stochastic properties. The advantages are that this filter is also applicable for higher-order systems, very easy to implement and uses one SP less than the UKF since the corresponding weight becomes zero. The last aspect is noteworthy because this first SP (as the mean of the set) may have a large negative weight. As a result, the error covariance matrix is not always guaranteed to be positive definite [7]. Nonetheless the CKF was newly derived as approximation of a Gaussian filter, it turned out to be a special case of the UKF with a certain choice of the filter parameters[5] (which is still remarkable). Refer to [6, 180] for a more detailed treatment.

[5] The UKF must be parametrized with $\alpha = 1$, $\beta = 0$ and $\kappa = 0$ in order to be like/resemble the CKF. Note that the $\alpha = 10^{-3}$, $\beta = 2$ and $\kappa = 0$ is usually recommended for Gaussian distributions [213].

Algorithm 2.3 (The Standard Cubature Kalman Filter)

Initialization step:

Initial state mean $$\hat{\boldsymbol{x}}_0^+ = E\{\boldsymbol{x}_0\} \tag{2.16a}$$

Initial covariance $$\mathbf{P}_0^+ = E\Big\{(\boldsymbol{x}_0 - \hat{\boldsymbol{x}}_0)(\boldsymbol{x}_0 - \hat{\boldsymbol{x}}_0)^{\mathrm{T}}\Big\} \tag{2.16b}$$

Sigma-point weights $$w_{\mathrm{m}} = (2n_x)^{-1}\,,\ w_{\mathrm{p}} = \sqrt{n_x} \tag{2.16c}$$

Prediction step:

New sigma-points $$\boldsymbol{\mathcal{X}}_{k-1}^+ = \widehat{\mathbf{X}}_{k-1}^+ + w_{\mathrm{p}}\left[\sqrt{\mathbf{P}_{k-1}^+}, -\sqrt{\mathbf{P}_{k-1}^+}\right] \tag{2.16d}$$

Propagated sigma-points $$\boldsymbol{\mathcal{X}}_k^* = \boldsymbol{f}(\boldsymbol{\mathcal{X}}_{k-1}^+, \boldsymbol{u}_{k-1}) \tag{2.16e}$$

Predicted state estimate $$\hat{\boldsymbol{x}}_k^- = w_{\mathrm{m}} \sum_{i=1}^{2n_x} \boldsymbol{\mathcal{X}}_k^*(:,i) \tag{2.16f}$$

Predicted covariance matrix $$\mathbf{P}_k^- = w_{\mathrm{m}} \boldsymbol{\mathcal{X}}_k^* (\boldsymbol{\mathcal{X}}_k^*)^{\mathrm{T}} - \hat{\boldsymbol{x}}_k^- (\hat{\boldsymbol{x}}_k^-)^{\mathrm{T}} + \mathbf{G}_{k-1}\mathbf{Q}_{k-1}\mathbf{G}_{k-1}^{\mathrm{T}} \tag{2.16g}$$

Correction step:

New sigma-points $$\boldsymbol{\mathcal{X}}_k^- = \widehat{\mathbf{X}}_k^- + w_{\mathrm{p}}\Big[\sqrt{\mathbf{P}_k^-}, -\sqrt{\mathbf{P}_k^-}\Big] \tag{2.16h}$$

Propagated sigma-points $$\boldsymbol{\mathcal{Y}}_k^* = \boldsymbol{h}(\boldsymbol{\mathcal{X}}_k^-, \boldsymbol{u}_k) \tag{2.16i}$$

Predicted output $$\hat{\boldsymbol{y}}_k = w_{\mathrm{m}} \sum_{i=1}^{2n_x} \boldsymbol{\mathcal{Y}}_k^*(:,i) \tag{2.16j}$$

Cross covariance matrix $$\mathbf{P}_k^{xy} = w_{\mathrm{m}} \boldsymbol{\mathcal{X}}_k^* \left(\boldsymbol{\mathcal{Y}}_k^*\right)^{\mathrm{T}} - \hat{\boldsymbol{x}}_k^- \left(\hat{\boldsymbol{y}}_k\right)^{\mathrm{T}} \tag{2.16k}$$

Innovation covariance matrix $$\mathbf{P}_k^{yy} = w_{\mathrm{m}} \boldsymbol{\mathcal{Y}}_k^* \left(\boldsymbol{\mathcal{Y}}_k^*\right)^{\mathrm{T}} - \hat{\boldsymbol{y}}_k \left(\hat{\boldsymbol{y}}_k\right)^{\mathrm{T}} + \mathbf{R}_k \tag{2.16l}$$

Kalman Gain $$\boldsymbol{\mathcal{K}}_k = \mathbf{P}_k^{xy} (\mathbf{P}_k^{yy})^{-1} \tag{2.16m}$$

Corrected covariance matrix $$\mathbf{P}_k^+ = \mathbf{P}_k^- - \boldsymbol{\mathcal{K}}_k \mathbf{P}_k^{yy} \boldsymbol{\mathcal{K}}_k^{\mathrm{T}} \tag{2.16n}$$

Filter Innovation $$\boldsymbol{v}_k = \boldsymbol{y}_k - \hat{\boldsymbol{y}}_k^- \tag{2.16o}$$

Corrected state estimate $$\hat{\boldsymbol{x}}_k^+ = \hat{\boldsymbol{x}}_k^- + \boldsymbol{\mathcal{K}}_k \boldsymbol{v}_k \tag{2.16p}$$

The standard CKF is summarized in Algorithm 2.3 (p. 24). The initialization is the same as for the linear filters. The initial covariance matrix $\mathbf{P}_0^+$ constitutes a measure of how trustworthy the initial state vector $\hat{\boldsymbol{x}}_0^+$ is.

The prediction step starts with the evaluation of the new SP in Eq. (2.16d). These points are determined as a symmetrical set $\boldsymbol{\mathcal{X}}_{k-1}^+$ from the previous corrected state estimate $\hat{\boldsymbol{x}}_{k-1}^+$ and the previous error covariance matrix $\mathbf{P}_{k-1}^+$. The matrix $\widehat{\mathbf{X}}_{k-1}^+ \in \mathbb{R}^{n_x \times 2n_x}$ contains $2n_x$ column vectors of $\hat{\boldsymbol{x}}_{k-1}^+$. Moreover, the term $\sqrt{\mathbf{P}_{k-1}^+} = \mathbf{S}_{k-1}^+$ is denoted as matrix square-root (MSR) of the covariance matrix which is computed efficiently using the Cholesky decomposition [206]. Therefore, the MSR is also termed as Cholesky factor. Secondly, Eq. (2.16e) is used to propagate all SP through the system's process model. Then, the new predicted state estimate $\hat{\boldsymbol{x}}_k^-$ is obtained with Eq. (2.16f) using these transformed SP. In a similar fashion, $\mathbf{P}_k^-$ is determined using Eq. (2.16g).

The correction step is conducted by assessing first $2n_x$ new SP with Eq. (2.16h) which are then propagated through the system's measurement model using Eq. (2.16i).[6] Then, Eq. (2.16j) computes the predicted measurement $\hat{\boldsymbol{y}}_k$, analogously to $\hat{\boldsymbol{x}}_k^-$, just with new SP $\boldsymbol{\mathcal{Y}}_k^*$. Afterwards, the Eqs. (2.16k) and (2.16l) compute the covariance matrices necessary to determine the current Kalman gain matrix $\boldsymbol{\mathcal{K}}_k$. Note that, Eqs. (2.16m) – (2.16p) are identical compared to the linear time-variant filter (cf. Algorithm 2.1).

Looking at Algoritm 2.3, the only free design parameters are the noise covariance matrices $\mathbf{Q}_k$ and $\mathbf{R}_k$. Thus, there are no additional degrees of freedom in design as with the UKF.

2.3.3 Square-Root Cubature Kalman Filter

The SR-CKF is a numerically more robust version of the CKF. According to [4], the advantages of using the square-root implementation are to improve the numerical accuracy, to enforce a double order precision, to preserve the properties of symmetry and positive definiteness as well as to facilitate the use of square-roots.

In practice, square-root implementations avoid effectively the two Cholesky decompositions for $\mathbf{P}_k^-$ and $\mathbf{P}_k^+$, defined by

$$\mathbf{P}_k^- = \mathbf{S}_k^- \left(\mathbf{S}_k^-\right)^{\mathrm{T}} \qquad \text{and} \qquad \mathbf{P}_k^+ = \mathbf{S}_k^+ \left(\mathbf{S}_k^+\right)^{\mathrm{T}} , \tag{2.17}$$

which are necessary in the CKF algorithm in Eq. (2.16d) and (2.16h). As a result, the SR-CKF shall enforce that the error covariances preserve their fundamental properties of symmetry and positive definiteness in each recursion step (despite the presence of numerical round-off errors and possibly too clean measurements). The complete set of equations for implemented square-root CKF is given in Algorithm 2.4 (p. 26).

The process and measurement noise covariances are incorporated in Eq. (2.16g) and (2.16l) using their Cholesky factors $\mathbf{S}_k^{\mathrm{Q}} = \sqrt{\mathbf{Q}_k}$ and $\mathbf{S}_k^{\mathrm{R}} = \sqrt{\mathbf{R}_k}$. The main differences between CKF and SR-CKF are observed in Eqs. (2.18g), (2.18l) and (2.18n). Therein, the QR factorization is used in order to obtain the covariance matrices. The function $\mathrm{qr}\{\cdot\}$ returns the upper triangular part $\mathbf{U}$ of the $\mathbf{A}$ matrix, thus $\mathbf{U} = \mathrm{qr}\{\mathbf{A}\}$ (cf. App. A.1.2 for more details).

One advantage of this SR-CKF and also the SR-CDKF is that no Cholesky updates and downdates on the matrix square-roots $\mathbf{S}_k^-$, $\mathbf{S}_k^+$ and $\mathbf{S}_k^{yy}$ are required, compared to the SR-UKF (cf. Algorithm A.1), since all the weights are guaranteed to be positive.

Moreover, the Kalman gain $\boldsymbol{\mathcal{K}}_k$ in Eq. (2.18m) is obtained directly from the relation

$$\boldsymbol{\mathcal{K}}_k \, \mathbf{S}_k^{yy} (\mathbf{S}_k^{yy})^{\mathrm{T}} = \mathbf{P}_k^{xy} \tag{2.19a}$$

$$\boldsymbol{\mathcal{K}}_k \, \mathbf{S}_k^{yy} = \mathbf{P}_k^{xy} / (\mathbf{S}_k^{yy})^{\mathrm{T}} \tag{2.19b}$$

$$\boldsymbol{\mathcal{K}}_k = \left(\mathbf{P}_k^{xy} / (\mathbf{S}_k^{yy})^{\mathrm{T}}\right) / \mathbf{S}_k^{yy} \tag{2.19c}$$

[6] Sometimes, the Eq. (2.16h) is skipped and $\boldsymbol{\mathcal{X}}_k^*$ is used in Eq. (2.16i) (instead of $\boldsymbol{\mathcal{X}}_k^-$). Thereby, the evaluation of the MSR $\mathbf{S}_k^-$ is omitted which mitigates the risk of divergence.

Algorithm 2.4 (The Square-Root Cubature Kalman Filter)

Initialization step:

Initial state estimate
$$\hat{\boldsymbol{x}}_0^+ = E\{\boldsymbol{x}_0\} \tag{2.18a}$$

Initial covariance MSR
$$\mathbf{S}_0^+ = \mathrm{chol}\Big\{E\Big\{(\boldsymbol{x}_0 - \hat{\boldsymbol{x}}_0)(\boldsymbol{x}_0 - \hat{\boldsymbol{x}}_0)^{\mathrm{T}}\Big\}\Big\}^{\mathrm{T}} \tag{2.18b}$$

Sigma-point weights
$$w_{\mathrm{m}} = (2n_x)^{-1},\ w_{\mathrm{p}} = \sqrt{n_x} \tag{2.18c}$$

Prediction step:

New sigma-points
$$\boldsymbol{\mathcal{X}}_{k-1}^+ = \widehat{\mathbf{X}}_{k-1}^+ + w_{\mathrm{p}}\begin{bmatrix}\mathbf{S}_{k-1}^+ & , -\mathbf{S}_{k-1}^+\end{bmatrix} \tag{2.18d}$$

Propagated sigma-points
$$\boldsymbol{\mathcal{X}}_k^* = \boldsymbol{f}(\boldsymbol{\mathcal{X}}_{k-1}^+, \boldsymbol{u}_{k-1}) \tag{2.18e}$$

Predicted state estimate
$$\hat{\boldsymbol{x}}_k^- = w_{\mathrm{m}} \sum_{i=1}^{2n_x} \boldsymbol{\mathcal{X}}_k^*(:, i) \tag{2.18f}$$

Predicted covariance MSR
$$\mathbf{S}_k^- = \mathrm{qr}\Big\{\Big[\sqrt{w_{\mathrm{m}}}\Big(\boldsymbol{\mathcal{X}}_k^* - \widehat{\mathbf{X}}_k^-\Big)\ ,\ \mathbf{S}_k^{\mathrm{Q}}\Big]\Big\}^{\mathrm{T}} \tag{2.18g}$$

Correction step:

New sigma-points
$$\boldsymbol{\mathcal{X}}_k^- = \widehat{\mathbf{X}}_k^- + w_{\mathrm{p}}\begin{bmatrix}\mathbf{S}_k^- & , -\mathbf{S}_k^-\end{bmatrix} \tag{2.18h}$$

Propagated sigma-points
$$\boldsymbol{\mathcal{Y}}_k^* = \boldsymbol{h}(\boldsymbol{\mathcal{X}}_k^-, \boldsymbol{u}_k) \tag{2.18i}$$

Predicted output
$$\hat{\boldsymbol{y}}_k = w_{\mathrm{m}} \sum_{i=1}^{2n_x} \boldsymbol{\mathcal{Y}}_k^*(:, i) \tag{2.18j}$$

Cross covariance matrix
$$\mathbf{P}_k^{xy} = w_{\mathrm{m}}\Big(\boldsymbol{\mathcal{X}}_k^- - \widehat{\mathbf{X}}_k^-\Big)\Big(\boldsymbol{\mathcal{Y}}_k^* - \widehat{\mathbf{Y}}_k\Big)^{\mathrm{T}} \tag{2.18k}$$

Innovation covariance MSR
$$\mathbf{S}_k^{yy} = \mathrm{qr}\Big\{\Big[\sqrt{w_{\mathrm{m}}}\Big(\boldsymbol{\mathcal{Y}}_k^* - \widehat{\mathbf{Y}}_k\Big)\ ,\ \mathbf{S}_k^{\mathrm{R}}\Big]\Big\}^{\mathrm{T}} \tag{2.18l}$$

Kalman Gain
$$\boldsymbol{\mathcal{K}}_k = \Big(\mathbf{P}_k^{xy} / (\mathbf{S}_k^{yy})^{\mathrm{T}}\Big) / \mathbf{S}_k^{yy} \tag{2.18m}$$

Corrected covariance MSR
$$\mathbf{S}_k^+ = \mathrm{qr}\Big\{\Big[\sqrt{w_{\mathrm{m}}}\Big(\boldsymbol{\mathcal{X}}_k^- - \widehat{\mathbf{X}}_k^-\Big) - \boldsymbol{\mathcal{K}}_k\Big(\boldsymbol{\mathcal{Y}}_k^* - \widehat{\mathbf{Y}}_k\Big)\ ,\ \boldsymbol{\mathcal{K}}_k\mathbf{S}_k^{\mathrm{R}}\Big]\Big\}^{\mathrm{T}} \tag{2.18n}$$

Filter Innovation
$$\boldsymbol{v}_k = \boldsymbol{y}_k - \hat{\boldsymbol{y}}_k^- \tag{2.18o}$$

Corrected state estimate
$$\hat{\boldsymbol{x}}_k^+ = \hat{\boldsymbol{x}}_k^- + \boldsymbol{\mathcal{K}}_k\boldsymbol{v}_k \tag{2.18p}$$

using an efficient two-step back-substitution algorithm, instead of a matrix inversion required in Eq. (2.16m). This trick is possible because the innovation error covariance MSR $\mathbf{S}_k^{yy}$ is square and has triangular shape.

In a nutshell, the SR-CKF exploits two powerful techniques, that are well-known from linear algebra, in order to improve the numerical stability: 1. the QR decomposition to assess the Cholesky factors and 2. the efficient least-squares to evaluate the Kalman gain efficiently.

2.4 Filter Performance Criteria

Generally speaking, there are several criteria to evaluate the estimation quality and the filter consistency (even without knowledge of the true state). In practice, it is crucial to assess the filter performance within simulation prior to application as well as during operation in the field. Performance measures are also required for KF with noise adaptation. This evaluation shall confirm the expected estimation quality and guarantee the requirements for practical use. In a nutshell, a good filter is considered as a consistent estimator which enforces the state estimate to be informative (with small covariances) [117].

The key performance indicator (KPI) for filter accuracy is essentially the state's estimation error (EE), given by

$$\boldsymbol{e}_k = \boldsymbol{x}_k - \hat{\boldsymbol{x}}_k^+ = \begin{bmatrix} e_{k,1} & e_{k,2} & \dots & e_{k,n_x} \end{bmatrix}^{\mathrm{T}} \tag{2.20}$$

which is originally defined as difference between the true state $\boldsymbol{x}_k$ and the posterior state estimate $\hat{\boldsymbol{x}}_k^+$. Again, n_x constitutes the dimension of the state, hence $\boldsymbol{e}_k \in \mathbb{R}^{n_x}$. The EE serves as the best possible indicator for filter performance when talking about estimation quality. Though, it is only available in simulation and not in real-world systems.

Alternative criteria are the measurement innovation $\boldsymbol{v}_k$ from Eq. (2.9) and the state residual $\widetilde{\boldsymbol{x}}_k$ from Eq. (2.10) which are available both in simulation and in reality (see Sect. 2.4.2). That is why, these quantities are exploited by adaptive filters rather than the EE that is applicable for simulation and design only.

Moreover, the Kalman gain (KG) and the covariance matrices can be analysed for performance self-assessment. The latter matrices constitute a measure of trust since smaller values suggest a more trustworthy estimate. The inverse of the covariance matrix is known as the information matrix, cf. [194] (p. 156 ff) for information filtering.

Consequently, the relevant criteria to judge the adaptive filter's performance are the magnitude of the residuals and their statistics [116].

2.4.1 Evaluation of the Estimation Error

The mean squared error (MSE) is used as an empirical statistical measure to evaluate the single state estimates $\hat{x}_{k,i}^+$ with $i \in [1, 2, \dots, n_x]$ for a specific time interval $k \in [0, 1, \dots, N–1]$. It is defined for the i-th state as

$$\mathrm{MSE}_i = \frac{1}{N} \sum_{k=0}^{N-1} e_{k,i}^2 \tag{2.21}$$

which is non-negative quality measure and ideally equal (or close) to 0. Besides the MSE, the normalized estimation error squared (NEES) is employed to check the consistency of the estimator. The definition writes

$$\varepsilon_{\mathrm{NS},k} = \boldsymbol{e}_k^{\mathrm{T}} \, \mathbf{M}_k^+ \, \boldsymbol{e}_k \tag{2.22}$$

where $\mathbf{M}_k^+$ is the matrix inverse of the covariance matrix $\mathbf{P}_k^+$. The NEES is supposed to follow a $\chi^2_{n_x}$ (chi-squared) distribution with n_x degrees of freedom when the estimator functions consistently [56, 40]. The definition of the average normalized estimation error squared (ANEES), cf. De Schutter et al. [189], is written as

$$\bar{\varepsilon}_{\text{NS}} = \frac{1}{n_x N} \sum_{k=0}^{N-1} \varepsilon_{\text{NS},k} \tag{2.23}$$

and is proposed to test whether an estimator should be rejected as non-credible or not. Ideally, $\bar{\varepsilon}_{\text{NS}}$ is close to 1 if the estimator is credible. The estimator is called optimistic in case of $\bar{\varepsilon}_{\text{NS}} \gg 1$ and called pessimistic for $\bar{\varepsilon}_{\text{NS}} \ll 1$. Note that there are also other measures, such as non-credibility indices (NCI), to assess the estimator performance.[7]

2.4.2 Evaluation of Innovation and State Residual

Since the EE is only known in simulation, the practitioner needs information that is actually available. Therefore, performance criteria based on innovation and state residual are essential to assess the performance online. First, the empirical mean value of the innovation sequence $\bar{\boldsymbol{v}}_k$ is considered as simple criterion. It is computed from the past N samples as follows:

$$\bar{\boldsymbol{v}}_k = \frac{1}{N} \sum_{j=j_0}^{k} \boldsymbol{v}_j \tag{2.24}$$

with $j_0 = k - N + 1$. Ideally, $\bar{\boldsymbol{v}}_k = \mathbf{0}$ must hold (for a sufficiently large number N of samples) to fulfil one condition for the optimality of the KF, namely that the innovation must be white and zero-mean [194]. Moreover, the normalized innovation squared (NIS)

$$\nu_{\text{NS},k} = \boldsymbol{v}_k^{\mathrm{T}} \, \mathbf{M}_k^{yy} \, \boldsymbol{v}_k \tag{2.25}$$

is employed as a second scalar measure. Therein, $\mathbf{M}_k^{yy}$ represents the inverse of the innovation covariance matrix $\mathbf{P}_k^{yy}$, not to be confused with the Cholesky factor $\mathbf{S}_k^{yy}$. The NIS is theoretically χ^2-distributed with n_y degrees of freedom in the ideal case, since $\boldsymbol{v}_k \in \mathbb{R}^{n_y}$. From Eq. (2.25), the summed normalized innovation squared (SNIS) is computed by adding up the NIS as follows:

$$\bar{\nu}_{\text{NS},k} = \sum_{j=j_0}^{k} \nu_{\text{NS},j} \ . \tag{2.26}$$

Furthermore, the innovation is used to compute an empirical covariance matrix

$$\hat{\mathbf{P}}_k^{yy} = \frac{1}{N} \sum_{j=j_0}^{k} \Big(\boldsymbol{v}_j - \bar{\boldsymbol{v}}_k\Big)\Big(\boldsymbol{v}_j - \bar{\boldsymbol{v}}_k\Big)^{\mathrm{T}} = \frac{1}{N} \boldsymbol{V}_{[j_0,k]} \boldsymbol{V}_{[j_0,k]}^{\mathrm{T}} - \bar{\boldsymbol{v}}_k \bar{\boldsymbol{v}}_k^{\mathrm{T}} \tag{2.27}$$

[7] Although the ANEES is an applicable measure, it exhibits a disadvantage with respect to the fact that optimism and pessimism are not equally weighted, see Rong Li et al. [119].

or considering $\bar{\boldsymbol{v}}_k = \mathbf{0}$ (which is true for the optimal filter design)

$$\hat{\mathbf{P}}_k^{yy} = \frac{1}{N}\sum_{j=j_0}^{k} \boldsymbol{v}_j \boldsymbol{v}_j^{\mathrm{T}} = \frac{1}{N}\boldsymbol{V}_{[j_0,k]}\boldsymbol{V}_{[j_0,k]}^{\mathrm{T}} \tag{2.28}$$

that is used as an estimate for filter adaptation [136, 3]. Therein, the auxiliary matrix $\boldsymbol{V}_{[j_0,k]}$ is defined as follows

$$\boldsymbol{V}_{[j_0,k]} = \begin{bmatrix} \boldsymbol{v}_{j_0} & \boldsymbol{v}_{j_0+1} & \boldsymbol{v}_{j_0+2} & \cdots & \boldsymbol{v}_k \end{bmatrix} \tag{2.29}$$

and thus contains the previous N innovation vectors columnwise (cf. App. A.1.4). Generally speaking, a consistent estimator should ensure that the empirical covariance estimate from Eq. (2.27) matches the filter's own (internal) confidence level from Eq. (2.8), such that

$$\hat{\mathbf{P}}_k^{yy} \approx \mathbf{P}_k^{yy} \tag{2.30}$$

is aimed at. This fact is (for instance) utilized as performance and optimization criterion in the master-slave adaptive filter where the diagonal elements of $\hat{\mathbf{P}}_k^{yy}$ are the measurements for the slave filter (cf. Sec. 2.5.2).

The second information to be used in a real filter is the state residual $\widetilde{\boldsymbol{x}}_k$. As can be seen from Eq. (2.10), the state residual is theoretically a zero-mean random variable

$$\bar{\widetilde{\boldsymbol{x}}}_k = \frac{1}{N}\sum_{j=j_0}^{k} \widetilde{\boldsymbol{x}}_j \tag{2.31}$$

for an optimal filter since then innovation $\boldsymbol{v}_k$ is zero-mean, too, with $j_0 = k - N + 1$.

Moreover, the normalized residual squared (NRS)

$$\tilde{x}_{\mathrm{NS},k}^2 = \widetilde{\boldsymbol{x}}_k^{\mathrm{T}}\, \widetilde{\mathbf{M}}_k\, \widetilde{\boldsymbol{x}}_k \tag{2.32}$$

where $\widetilde{\mathbf{M}}_k$ is the inverse of the state residual covariance matrix $\widetilde{\mathbf{P}}_k$, see (2.40) or (2.7b).

Furthermore, the state residual is used to compute an empirical covariance matrix

$$\hat{\bar{\mathbf{P}}}_k = \frac{1}{N}\sum_{j=j_0}^{k} \Big(\widetilde{\boldsymbol{x}}_j - \bar{\widetilde{\boldsymbol{x}}}_k\Big)\Big(\widetilde{\boldsymbol{x}}_j - \bar{\widetilde{\boldsymbol{x}}}_k\Big)^{\mathrm{T}} = \frac{1}{N}\widetilde{\boldsymbol{X}}_{[j_0,k]}\widetilde{\boldsymbol{X}}_{[j_0,k]}^{\mathrm{T}} - \bar{\widetilde{\boldsymbol{x}}}_k\bar{\widetilde{\boldsymbol{x}}}_k^{\mathrm{T}} \tag{2.33}$$

or considering $\bar{\widetilde{\boldsymbol{x}}}_k = \mathbf{0}$ (for the ideal case)

$$\hat{\bar{\mathbf{P}}}_k = \frac{1}{N}\sum_{j=j_0}^{k} \widetilde{\boldsymbol{x}}_j\widetilde{\boldsymbol{x}}_j^{\mathrm{T}} = \frac{1}{N}\widetilde{\boldsymbol{X}}_{[j_0,k]}\widetilde{\boldsymbol{X}}_{[j_0,k]}^{\mathrm{T}} \tag{2.34}$$

that is used as an estimate for filter adaptation [136, 3]. Therein, the auxiliary matrix $\widetilde{\boldsymbol{X}}_{[j_0,k]}$ is defined as follows

$$\widetilde{\boldsymbol{X}}_{[j_0,k]} = \begin{bmatrix} \widetilde{\boldsymbol{x}}_{j_0} & \widetilde{\boldsymbol{x}}_{j_0+1} & \widetilde{\boldsymbol{x}}_{j_0+2} & \cdots & \widetilde{\boldsymbol{x}}_k \end{bmatrix} \tag{2.35}$$

which contains the previous N state residual vectors columnwise.

In summary, the empirical covariance matrices in Eq. (2.27) and (2.33) are crucial as measures for filter optimality in order to adjust noise covariance matrices automatically. Thus, these will appear again in Sect. 2.5 as input quantities to the adaptive filter. Moreover also for standard filters, these measures serve as performance criteria to evaluate possible defects online.

2.4.3 Evaluation of Error Covariance Matrices

As stated before, the covariance matrix represents another performance measure to assess the uncertainty of the estimated state $\hat{\boldsymbol{x}}_k$, see Eq. (2.6b). Therefore, the filter provides simultaneously its own confidence measures in form of covariance matrices. It is remarkable that these quantities are all connected by one covariance update function

$$\mathbf{P}_k^+ = \mathbf{P}_k^- - \boldsymbol{\mathcal{K}}_k \mathbf{P}_k^{yy} \boldsymbol{\mathcal{K}}_k^{\mathrm{T}} \tag{2.36}$$

which is only parametrized by the Kalman gain. This is notably similar compared to state's update Eq. (2.7a). Rearranging Eq. (2.36) and presupposing a positive definite covariance matrix $\mathbf{P}_k^{yy} \succ 0$, cf. Eq. (2.8) or (2.16m), then yields

$$\mathbf{P}_k^- - \mathbf{P}_k^+ = \boldsymbol{\mathcal{K}}_k \mathbf{P}_k^{yy} \boldsymbol{\mathcal{K}}_k^{\mathrm{T}} \succeq 0 \tag{2.37}$$

which means that the uncertainty after the correction step is always smaller or equal than the one from the prediction step. The degree of reduction provides additional insight into the filter performance in such a way that it shows how much confidence the filter thinks it has gained in its state estimate $\hat{\boldsymbol{x}}_k^+$ from the correction step itself. This depends strongly on the Kalman gain $\boldsymbol{\mathcal{K}}_k$.

One aspect of the KF's covariances (the practitioner must be always aware of in general) is that the matrices $\mathbf{P}_k^-$, $\mathbf{P}_k^+$ and $\mathbf{P}_k^{yy}$ are computed completely independently of the actual measurements $\hat{\boldsymbol{y}}_k$. This can be understood best in Algorithm 2.1 comparing Eqs. (2.11d), (2.11f) to (2.11i). As long as the model and the noise assumptions are correct, this is not critical. Though, it makes the filter vulnerable in case the assumptions do not hold because the filter might *think* it works perfectly, but it actually does not. In order to address this issue, adaptive filters are used which tackle the deficiency effectively (cf. Sect. 2.5).

Now, how can the covariances be used to assess the filter performance? To answer the question, consider the following items:

- Analyse the convergence: The error covariances should approach (approximately) constant values after a transition time. This is an indicator whether the filter converges or not. Though, that does not necessarily mean convergence to the correct value [189].

- Analyse the diagonal elements: A simple approach is to look at the diagonal elements of the error covariances

$$\boldsymbol{p}_k^- = \operatorname{diag}\left\{\mathbf{P}_k^-\right\} = \begin{bmatrix} p_{k,11}^- & p_{k,22}^- & \cdots & p_{k,ii}^- & \cdots & p_{k,n_x n_x}^- \end{bmatrix}^{\mathrm{T}} \tag{2.38a}$$

$$\boldsymbol{p}_k^+ = \operatorname{diag}\left\{\mathbf{P}_k^+\right\} = \begin{bmatrix} p_{k,11}^+ & p_{k,22}^+ & \cdots & p_{k,ii}^+ & \cdots & p_{k,n_x n_x}^+ \end{bmatrix}^{\mathrm{T}} \tag{2.38b}$$

  The square-roots $\sqrt{p_{k,ii}^-}$ and $\sqrt{p_{k,ii}^+}$ provide a simple (but possibly biased, if $\mathbf{P}_k^-$ and $\mathbf{P}_k^+$ are far off diagonal) estimate for the 1σ confidence region around the i-th state estimate.

- Analyse the singular values: A more appropriate measure is to assess the singular values σ_i, SV for short, of the error covariances obtained by singular value decomposition (SVD)[8]. Though the SV provide mathematically a suitable measure, still any conclusion about the accuracy of the estimate based on either the diagonal elements or the singular values must be drawn very carefully. This is due to the further above mentioned deficiency of KF's covariance propagation.

- Consider the error ellipses: Based on the SVD or singular values σ_i, an uncertainty ellipsoid can be plotted in the state space around the estimate. Such ellipsoids are expressed mathematically by

$$(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k^-)^{\mathrm{T}} \left(\mathbf{P}_k^-\right)^{-1} (\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k^-) = \chi^2 \tag{2.39a}$$

$$(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k^+)^{\mathrm{T}} \left(\mathbf{P}_k^+\right)^{-1} (\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k^+) = \chi^2 \tag{2.39b}$$

  which collapse to uncertainty ellipses in the 2D state space (cf. Fig. 2.1). Note the resemblance of Eqs. (2.39) to the definition of the NEES in Eq. (2.22). In order to assess these ellipses numerically, a Gaussian distribution with a known covariance $\mathbf{P}_k^+$ is implied. First, the SVD is used to assess its singular values and then χ^2 as scaling parameter is determined based on the dimension n_x and the probability p (cf. [93]).

In conclusion, the covariance matrices constitute another elaborate measure for the uncertainty analysis of the estimation process. These matrices are used to compute uncertainty regions, displayed as ellipsoids in the state space. Though as the example in Fig. 2.1 impressively shows, these measures are possibly heavily flawed.

2.4.4 Comparison of the Filter's and the Empirical Covariances

For this reason, not only the filter's covariances have to be assessed but also the empirical covariances determined from the state estimates and the innovation. This is the only way in practical terms to tackle/circumvent effectively the deficiency.

[8] Refer to Sect. A.1.3 for more details about the SVD. Note that, the σ_i are not equal to the standard deviation σ but rather correspond to the variance σ^2 of the random variables.

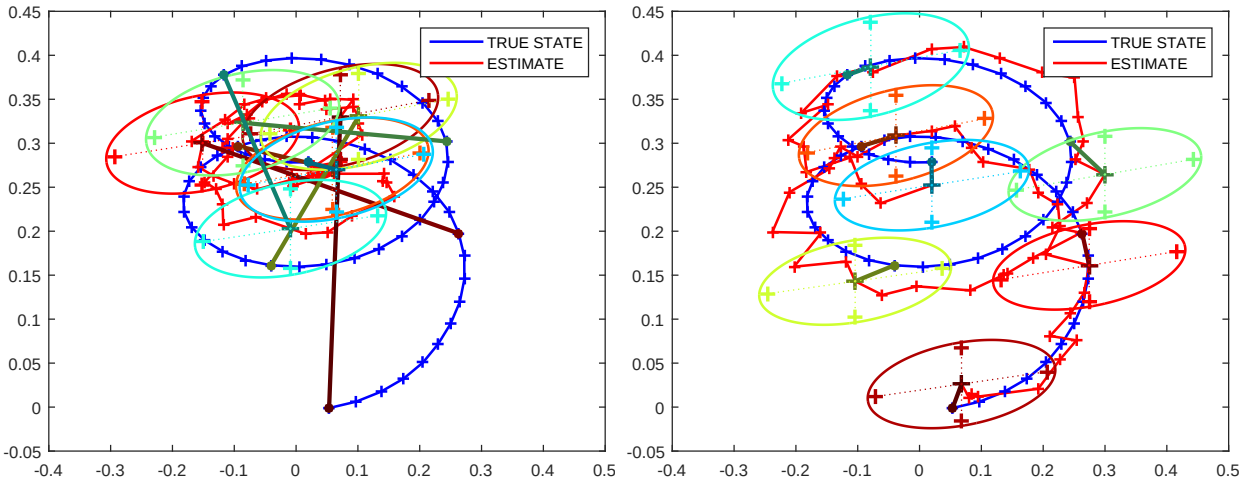

Figure 2.1: Illustrative comparison of the true and estimated states for two different filter designs (left: poor design, right: optimal design). The plots include in each case the corrected sigma-points and also the uncertainty ellipses, cf. Eq. (2.39b) with $p = 0.95$, $n_x = 2$ and $k = 2.448$, for a confidence region of 95% (considering only every 10-th sample for the sake of clarity). All uncertainty ellipses have approximately the same size since both filters have already converged. For the optimal design, all true states are located within the ellipses, thus the estimates are informative and consistent. Contrary, the poorly designed filter does not recognize how bad the estimates actually are.

The actual performance is measured by two uncertainty measures based on empirical computation. These are the empirical state residual covariance matrix $\hat{\widetilde{\mathbf{P}}}_k$, defined in Eq. (2.27), and the empirical innovation covariance matrix $\hat{\mathbf{P}}_k^{yy}$, defined in Eq. (2.33).

The predicted performance (also denoted as the filter's self-performance) is assessed by the state's residual covariance matrix, defined by

$$\widetilde{\mathbf{P}}_k = \boldsymbol{\mathcal{K}}_k \mathbf{P}_k^{yy} \boldsymbol{\mathcal{K}}_k^{\mathrm{T}} = \mathbf{P}_k^{-} - \mathbf{P}_k^{+} \ , \tag{2.40}$$

cf. Eq. (2.36), and the innovation covariance matrix $\mathbf{P}_k^{yy}$ which are both provided by the filter once in every recursion step. Note that, the negative sign in Eq. (2.40) is different compared to the definition of the state residual $\widetilde{\boldsymbol{x}}_k$ in Eq. (2.10). Rearranging Eq. (2.40), yields

$$\mathbf{P}_k^{yy} = \boldsymbol{\mathcal{K}}_k^{\#} \widetilde{\mathbf{P}}_k \left(\boldsymbol{\mathcal{K}}_k^{\#} \right)^{\mathrm{T}} \tag{2.41}$$

where $\boldsymbol{\mathcal{K}}_k^{\#} \in \mathbb{R}^{n_y \times n_x}$ is the Moore-Penrose inverse [10] of the master filter's Kalman gain $\boldsymbol{\mathcal{K}}_k \in \mathbb{R}^{n_x \times n_y}$, defined by

$$\boldsymbol{\mathcal{K}}_k^{\#} = \left(\boldsymbol{\mathcal{K}}_k^{\mathrm{T}} \boldsymbol{\mathcal{K}}_k \right)^{-1} \boldsymbol{\mathcal{K}}_k^{\mathrm{T}} \ . \tag{2.42}$$

The above equations are also used in the different adaptation rules for filter parameter update.

2.5 Adaptive Kalman Filters

The optimality of the Kalman filter is only provided when the dynamic model and the noise statistics are accurately known in advance. Otherwise the Kalman filter theoretically must not be applied. Though in practical terms, the model is always an approximation of the real system. Moreover, the statistics of the process and measurement noise are also vague and not completely known quantities [130]. That is why the latter are often interpreted as design parameters of the filter.

Still, the practitioner must ensure that both the model and the statistics are as close as possible to their real/optimal values to avoid filter divergence and to reach the prescribed performance. This is especially true in the presence of time-varying parameters as well as initially unknown/poorly known parameters. In general, there are two types of adaptive filtering to be distinguished:

- Model parameter adaptation: This concept addresses the mis-modelling between the real system and the underlying internal filter model. Since the model-based algorithms rely on preferably accurate models, it is important to make sure that deviations remain small. These deviations may result from deliberate simplifications or time-varying parameters (for instance, due to wear). If the root-cause can be associated with certain model parameters, the problem can either be tackled using a standard SPKF (where model parameters are augmented as constant states) or, alternatively, using a separate parameter estimation algorithm [122, 84].
- Filter parameter adaptation: This type addresses the mismatches between the designed filter noise parameters and the actual noise environment in the real-world system. Contrary to the state or parameter estimation, the filter adaptation seeks to match the underlying noise statistics by adapting either the process or the measurement noise (or occasionally the Kalman gain). This means that a set of originally fixed design parameters is released for adaptation.

The remainder of this section provides a brief overview about the available adaptation rules and presents two concepts in greater detail: The master-slave approach and the maximum likelihood estimation (MLE).

2.5.1 Introduction to Filter Adaptation Rules

Although the standard SPKF tackles the problem of system nonlinearity effectively, it cannot address the problem of changing and/or unknown noise statistics by nature (cf. Fig. 2.2a). Lack of knowledge about these noise statistics is often present in technical systems. Unfortunate design may lead to filter divergence or severe performance degradation in practice.

One way to deal with this issue is to go with a very conservative design of the static noise covariance matrices $\mathbf{Q}_\mathrm{d}$ and $\mathbf{R}_\mathrm{d}$ which is though not always very appealing.

As discussed in Sect. 2.4.3, this drawback is directly related to the inability of the Kalman filter to utilize the new measurement information $\boldsymbol{y}_k$ on the level of covariance propagation. That means that noise adaptive filters not only address the first-order statistical moment (the mean), like in the standard SPKF, but also the second-order moment statistical information (the covariance) [136].

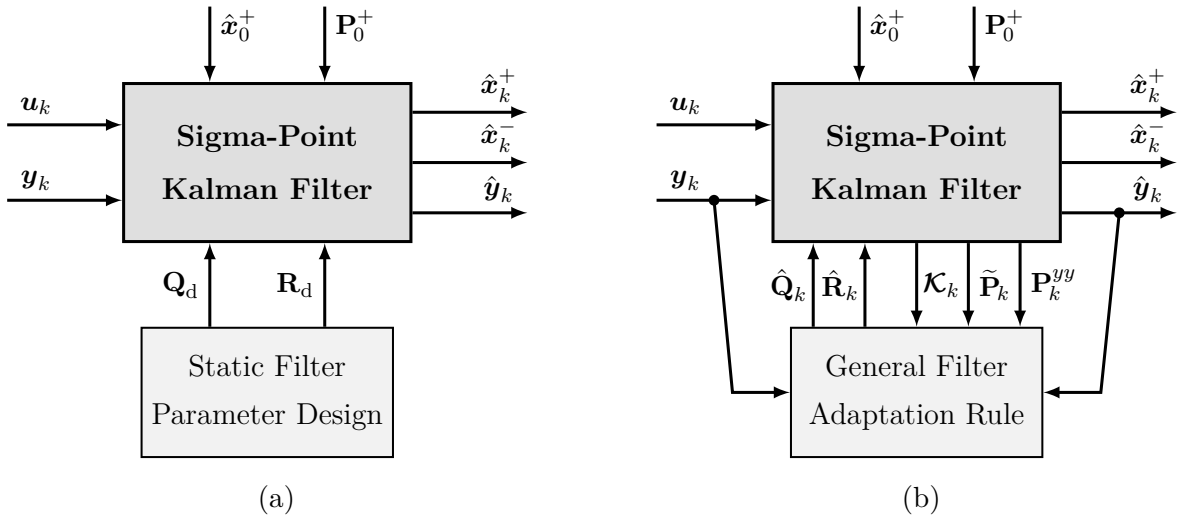

Figure 2.2: Schematic structures of the standard SPKF with static filter design (a) and the noise adaptive SPKF with a general innovation-based adaptation rule (b)

Generally, one must distinguish between the following two fundamental concepts [136]:

- **Innovation-based adaptive estimation** (IAE): The algorithms, that fall into the class of IAE, usually adapt the design of noise covariance matrices. Sometimes, the Kalman gain $\boldsymbol{\mathcal{K}}_k$ is directly adjusted [131, 130]. The basis for the adaptation constitutes the innovation sequence itself or its (theoretical) whiteness property.
- **Multiple model adaptive estimation** (MMAE): As the name implies, the algorithm works with a bank of several parallel estimators (see Fig. 2.3). Each of the K estimators runs with a (slightly) different model and/or noise assumption where $\boldsymbol{\vartheta}_j$ denotes the varied/selected noise parameters $\boldsymbol{\vartheta}$ for the j-th Kalman filter. The MMAE produce in total K different estimated state vectors in every recursion step. The joint adaptive estimate is then evaluated by a weighted sum

$$\hat{\boldsymbol{x}}_k^+ = \sum_{j=1}^{K} p(\boldsymbol{\vartheta}_j|\boldsymbol{y}_k)\, \hat{\boldsymbol{x}}_k^+(\boldsymbol{\vartheta}_j) \tag{2.43}$$

  wherein $p(\boldsymbol{\vartheta}_j|\boldsymbol{y}_k)$ is the probability of each state estimate $\hat{\boldsymbol{x}}_k^+(\boldsymbol{\vartheta}_j)$, given all measurements up to $\boldsymbol{y}_k$. The probability is evaluated based on innovation $\boldsymbol{v}_k(\boldsymbol{\vartheta}_j)$ or state residual $\widetilde{\boldsymbol{x}}_k(\boldsymbol{\vartheta}_j)$ of all filters [136].

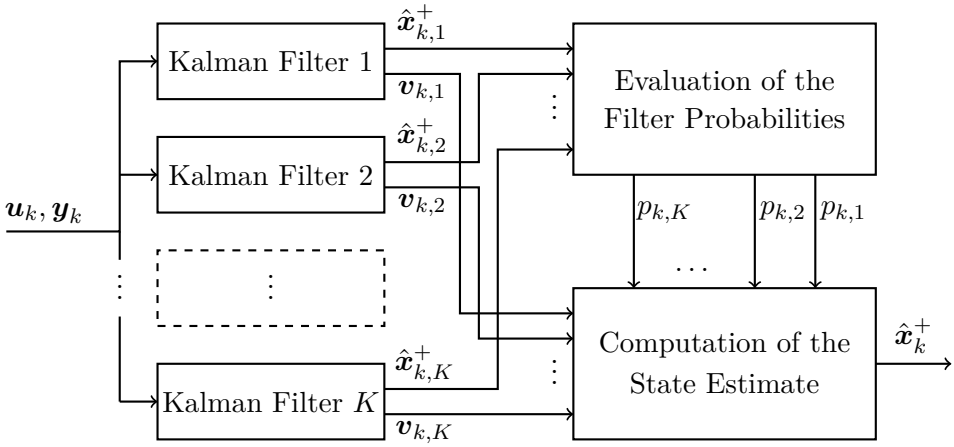

Figure 2.3: Schematic structure of multiple model adaptive estimation which is realized using a bank of K parallel Kalman filters [204, 184]

Due to the exponentially growing number of filters with increasing dimension of $\boldsymbol{\vartheta}$, the MMAE are not suited for higher dimensions and critical sample times. A typical application, where these filter banks are used, is for online failure detection [67, 64], working particularly well if the failure is known in advance.

In the following, the IAE are in the focus.[9] A literature research has revealed a great variety of such IAE methods that allow to estimate the underlying noise statistics. An overview is provided in [130, 65]. In the following, only the most important of them are presented (since no claim to completeness is implied):

- **Maximum Likelihood Estimation** (MLE): The MLE is one of the earliest approaches used for adaptive Kalman filtering, originally proposed by Mehra [131]. In MLE, a likelihood function $L(\boldsymbol{x}_k, \boldsymbol{\vartheta})$ is defined to obtain the maximum likelihood estimates $\hat{\boldsymbol{\vartheta}}$ assuming independent and identically distributed measurements. Theoretically, both $\mathbf{Q}_k$ and $\mathbf{R}_k$ can be adjusted at the same time [136] (p. 197), though this is rarely feasible in practice. The MLE has been adopted by several authors in the past [127, 136, 210, 153, 36, 200]. The most popular method among them still stems from Maybeck [127]. Refer to Sect. 2.5.3 for more details.

- **Correlation Methods**: The underlying idea of this method is to correlate the output $\boldsymbol{y}_k$ or the innovation $\boldsymbol{v}_k$ [131, 129]. Since the innovation $\boldsymbol{v}_k$ is less correlated than the measurement $\boldsymbol{y}_k$, the innovation based approach is more efficient. An advantage of the output correlation is that no prior values for $\mathbf{Q}_k$ and $\mathbf{R}_k$ must be available which qualifies the correlation method as a start-up procedure. A major drawback is its restriction to linear time-invariant systems which disqualifies the method pretty much for nonlinear control systems [130].

[9] Refer to the publications [124, 118, 66, 136, 67] for further details on MMAE.

- **Covariance Matching**: This technique has the goal to make the innovation (residuals) consistent with the filter-computed covariance [130, 33]. If the process noise is known beforehand, there exists a closed-form solution (here exemplarily for the standard CKF)

$$\hat{\mathbf{R}}_{k+1} = \frac{1}{N}\sum_{j=0}^{N-1} \boldsymbol{v}_{k-j}\boldsymbol{v}_{k-j}^{\mathrm{T}} - \left[w_{\mathrm{m}}\boldsymbol{\mathcal{Y}}_k^*\left(\boldsymbol{\mathcal{Y}}_k^*\right)^{\mathrm{T}} - \hat{\boldsymbol{y}}_k\left(\hat{\boldsymbol{y}}_k\right)^{\mathrm{T}}\right], \tag{2.44}$$

  cf. Eq. (2.16l), to estimate the measurement noise. Contrary with known $\mathbf{R}_k$, there exists only a unique solution for the process noise $\mathbf{Q}_k$, if $n_x = n_y$ holds true. Otherwise, the number of unknowns in $\mathbf{Q}_k$ must be restricted. The convergence of the CM technique is in general doubted for estimating $\mathbf{Q}_k$ and $\mathbf{R}_k$ [130, 141].

- **MIT Rule**: Furthermore, the so-called MIT rule [200, 65] is a rather complicated computation which is also error-prone from an implementation stand point and numerical stability is hard to evaluate. In the MIT-SPKF, the cost function

$$J_k = \mathrm{Tr}\left\{\left(\mathbf{P}_k^{yy} - \hat{\mathbf{P}}_k^{yy}\right)^2\right\} \tag{2.45}$$

  is proposed in order to minimize the difference between the filter-computed and the actual innovation covariance [65], cf. Eq. (2.16l) and (2.27). As can be understood, the MIT algorithm requires the evaluation of the partial derivatives for Eq. (2.45) with respect to the sigma-points. This requires a tricky calculation and also a heavy computational burden [65].

- **Master-Slave Filter Approach** (MS): More recently, the master-slave approach has been introduced [201, 199]. The idea is to use a second linear KF or SPKF in parallel (to the master filter) in order to estimate the noise variables and update them alternately to the states. The main advantages are the reduced complexity, the ease of implementation and the flexibility to choose only the critical noise parameters. Among a few others, these benefits make the MS filters a competitive alternative to the established adaptation rules. Refer to Sect. 2.5.2 and [125] for a more detailed introduction.

In general, the main critical aspect of the adaptive algorithms is the filter convergence since it cannot be guaranteed or proven mathematically (due to the feedback character of the algorithm). This said, especially the joint estimation of process and measurement noise at a time is doubtful and reduces the robustness. Therefore, it should be avoided or handled carefully.

In consequence, two steps are in particular crucial for the practical implementation:

1. The control engineer must take responsibility for the configuration and design of the slave algorithm. He or she must use the available expert knowledge to avoid in any case non-identifiable settings and allow to adapt only the relevant noise parameters and refrain from changing the ones with negligible effect.

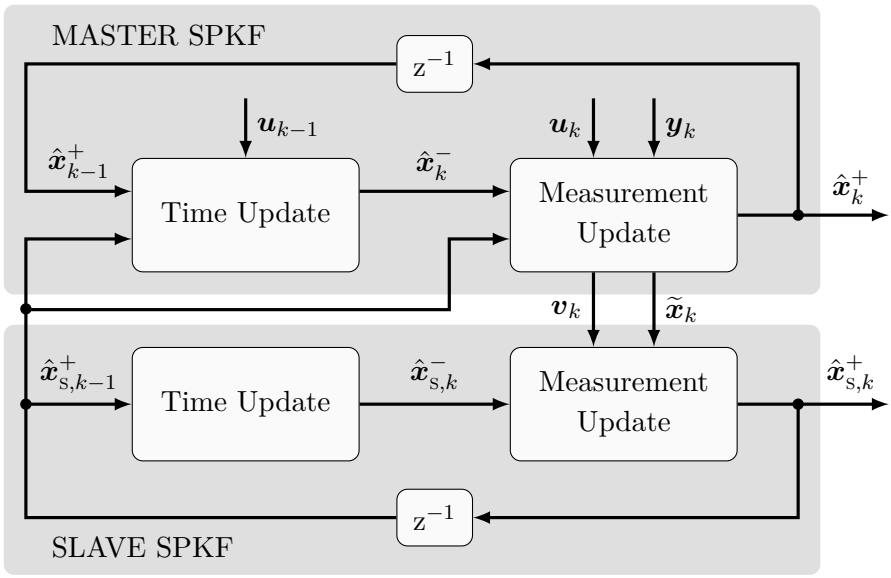

Figure 2.4: Generic structure of the master-slave SPKF

2. An exhaustive simulation study must be conducted prior to implementation. This needs to be done in a highly automated fashion to assess every possible worst-case scenario in advance. Only then, the filter is allowed to run on a control hardware device.

Based on the results of a preliminary study [184], only the two most promising approaches have been selected as representatives for adaptive filters. These are the master-slave approach and the MLE which are both introduced hereafter.

2.5.2 Master-Slave Approach

The master-slave algorithm has been introduced by Song et al. [201] in 2007 as an extension for the standard UKF. The authors proposed a configuration of two parallel UKFs where the first filter estimates the states $\hat{\boldsymbol{x}}_k^+$ and the second filter estimates the released noise parameters $\boldsymbol{\vartheta}_k^+$, as shown in Fig. 2.4. Later, the same group of researchers published a simplified algorithm using a simple KF as the slave filter [162, 199]. This has been a reasonable step since there is often neither the noise statistics nor their evolution over time known previously. Not to mention that there exists an idea of how such a nonlinear noise model could look like.

In the following, the ideas are generalized for separate process and measurement noise estimation using either the innovation-based or the residual-based approach. The necessary assumption behind this approach is that solely the diagonal elements of the covariance matrices matter. The off-diagonal elements of the estimated process or measurement noise are

set to zero.[10] Moreover, the off-diagonal elements of the estimation error and innovation covariances matrices are completely ignored. Especially, the latter should be checked in practice whether it holds true or not.

A summary of the developed system of equations is given in Algorithm 2.5. Therein, the slave filter is perceived as a linear time-variant KF with a special choice of the system matrices, cf. Eqs. (2.46). Noticeably, only the output matrix $\mathbf{C}_{\mathrm{s},k}$ is variable and depends on the choice of possible options. The matrices $\mathbf{A}_{\mathrm{s},k} = \mathbf{I}$, $\mathbf{B}_{\mathrm{s},k} = \mathbf{0}$ and $\mathbf{D}_{\mathrm{s},k} = \mathbf{I}$, respectively, are constant entities.[11]

In Algorithm 2.5, choose either option 1 for process noise adaptation or option 2 for measurement noise adaptation, see Eqs. (2.47). Then, choose either option A in order to use the innovation-based approach or option B for the residual-based approach. This essentially defines the *measurements* used in the slave filter (as the diagonal elements of the innovation covariance $\mathbf{P}_k^{yy}$ or the residual covariance $\widetilde{\mathbf{P}}_k$).

As a side note, consider Eq. (2.46e) as the most noteworthy of Algo. 2.5. It constitutes the (linear) observation model of the slave filter. The slave's control input vector $\boldsymbol{u}_{\mathrm{s},k}$ contains only external quantities provided by the master algorithm (covariance matrices and/or the Kalman gain, depending on the above options). The state vector $\boldsymbol{x}_{\mathrm{s},k}$ contains the selected noise parameters.

The free design parameters of the slave filter are the covariance matrices $\mathbf{Q}_{\mathrm{s},k}$ and $\mathbf{R}_{\mathrm{s},k}$. These affect in practice the allowable rates of change of the covariance $\mathbf{Q}_k$ or $\mathbf{R}_k$. Their dimensions change with respect to the considered options. Moreover, the window length N to evaluate the empirical covariance matrices in Eqs. (2.47b), (2.47e), (2.47h) and (2.47k) must be chosen appropriately. Refer to Sect. 4.4 for more details on the design methodology.

Concluding, the main advantages, advocating for the proposed slave filter, are

- the reduced computational complexity,
- the freely selectable noise variables that ought to be adapted,
- the ease of implementation,
- the usability of master and slave filter in one filter class, and
- the same design methodology as for the master filter.

[10] Note that even with a diagonal spectral density matrix $\mathbf{Q}_{\mathrm{c}}(t)$ the corresponding covariance matrix $\mathbf{Q}_k$ is in general not necessarily diagonal which is due to the sampling process (see Stengel [203], p. 328 ff). Though for sufficiently small sample times T_{s}, this can be assumed since then $\mathbf{Q}_k \approx \mathbf{Q}_{\mathrm{c}}(t_k)T_{\mathrm{s}}$ holds true.

[11] Refer to App. A.5 for details on the derivation of these equations. The symbol S labels in this thesis the variables and parameters related to the slave filter. In Algorithm 2.5, the operator $\mathrm{diag}\{\cdot\}$ is applied to, both, create a diagonal matrix from a given column vector and as well to extract the diagonal elements of a square matrix (as a column vector). The symbol $\circ$ indicates an element-wise matrix multiplication.

Algorithm 2.5 (Discrete-time Slave Kalman filter)

Initialization step:

$$\text{Initial State Mean} \qquad \hat{\boldsymbol{x}}^{+}_{\mathrm{s},0} = E\{\boldsymbol{x}_{\mathrm{s},0}\} \tag{2.46a}$$

$$\text{Initial State Covariance} \qquad \mathbf{P}^{+}_{\mathrm{s},0} = E\Big\{(\boldsymbol{x}_{\mathrm{s},0} - \hat{\boldsymbol{x}}_{\mathrm{s},0})(\boldsymbol{x}_{\mathrm{s},0} - \hat{\boldsymbol{x}}_{\mathrm{s},0})^{\mathrm{T}}\Big\} \tag{2.46b}$$

Prediction step:

$$\text{Predicted state} \qquad \hat{\boldsymbol{x}}^{-}_{\mathrm{s},k} = \hat{\boldsymbol{x}}^{+}_{\mathrm{s},k-1} \tag{2.46c}$$

$$\text{Predicted error covariance} \qquad \mathbf{P}^{-}_{\mathrm{s},k} = \mathbf{P}^{+}_{\mathrm{s},k-1} + \mathbf{Q}_{\mathrm{s},k} \tag{2.46d}$$

$$\text{Predicted output} \qquad \hat{\boldsymbol{y}}_{\mathrm{s},k} = \mathbf{C}_{\mathrm{s},k}\hat{\boldsymbol{x}}^{-}_{\mathrm{s},k} + \boldsymbol{u}_{\mathrm{s},k} \tag{2.46e}$$

Correction step:

$$\text{Cross Covariance Matrix} \qquad \mathbf{P}^{xy}_{\mathrm{s},k} = \mathbf{P}^{-}_{\mathrm{s},k}\mathbf{C}^{\mathrm{T}}_{\mathrm{s},k} \tag{2.46f}$$

$$\text{Innovation Covariance Matrix} \qquad \mathbf{P}^{yy}_{\mathrm{s},k} = \mathbf{C}_{\mathrm{s},k}\mathbf{P}^{-}_{\mathrm{s},k}\mathbf{C}^{\mathrm{T}}_{\mathrm{s},k} + \mathbf{R}_{\mathrm{s},k} \tag{2.46g}$$

$$\text{Kalman Gain} \qquad \boldsymbol{\mathcal{K}}_{\mathrm{s},k} = \mathbf{P}^{xy}_{\mathrm{s},k}(\mathbf{P}^{yy}_{\mathrm{s},k})^{-1} \tag{2.46h}$$

$$\text{Corrected Error Covariance} \qquad \mathbf{P}^{+}_{\mathrm{s},k} = \mathbf{P}^{-}_{\mathrm{s},k} - \boldsymbol{\mathcal{K}}_{\mathrm{s},k}\mathbf{P}^{yy}_{\mathrm{s},k}\boldsymbol{\mathcal{K}}^{\mathrm{T}}_{\mathrm{s},k} \tag{2.46i}$$

$$\text{Corrected state estimate} \qquad \hat{\boldsymbol{x}}^{+}_{\mathrm{s},k} = \hat{\boldsymbol{x}}^{-}_{\mathrm{s},k} + \boldsymbol{\mathcal{K}}_{\mathrm{s},k}\boldsymbol{v}_{\mathrm{s},k} = \hat{\boldsymbol{x}}^{-}_{\mathrm{s},k} + \boldsymbol{\mathcal{K}}_{\mathrm{s},k}(\boldsymbol{y}_{\mathrm{s},k} - \hat{\boldsymbol{y}}_{\mathrm{s},k}) \tag{2.46j}$$

Opt. 1A: Process noise estimation $\hat{\mathbf{Q}}_k = \mathrm{diag}\{\hat{\boldsymbol{x}}^{+}_{\mathrm{s},k}\}$ using the innovation-based approach

$$\text{Control input} \qquad \boldsymbol{u}_{\mathrm{s},k} = \mathrm{diag}\Big\{\boldsymbol{\mathcal{K}}^{\#}_{k}(\widetilde{\mathbf{P}}_k - \hat{\mathbf{Q}}_{k-1})(\boldsymbol{\mathcal{K}}^{\#}_{k})^{\mathrm{T}}\Big\} \quad \Rightarrow \text{cf. Eq. (2.41)} \tag{2.47a}$$

$$\text{Measured output} \qquad \boldsymbol{y}_{\mathrm{s},k} = \mathrm{diag}\{\hat{\mathbf{P}}^{yy}_{k}\} \quad \Rightarrow \text{cf. Eq. (2.27)} \tag{2.47b}$$

$$\text{Output matrix} \qquad \mathbf{C}_{\mathrm{s},k} = \boldsymbol{\mathcal{K}}^{\#}_{k} \circ \boldsymbol{\mathcal{K}}^{\#}_{k} \quad \Rightarrow \text{cf. Eq. (2.42)} \tag{2.47c}$$

Opt. 1B: Process noise estimation $\hat{\mathbf{Q}}_k = \mathrm{diag}\{\hat{\boldsymbol{x}}^{+}_{\mathrm{s},k}\}$ using the residual-based approach

$$\text{Control input} \qquad \boldsymbol{u}_{\mathrm{s},k} = \mathrm{diag}\Big\{\widetilde{\mathbf{P}}_k - \hat{\mathbf{Q}}_{k-1}\Big\} \tag{2.47d}$$

$$\text{Measured output} \qquad \boldsymbol{y}_{\mathrm{s},k} = \mathrm{diag}\{\hat{\widetilde{\mathbf{P}}}_k\} \quad \Rightarrow \text{cf. Eq. (2.33)} \tag{2.47e}$$

$$\text{Output matrix} \qquad \mathbf{C}_{\mathrm{s},k} = \mathbf{I} \tag{2.47f}$$

Opt. 2A: Measurement noise estimation $\hat{\mathbf{R}}_k = \mathrm{diag}\{\hat{\boldsymbol{x}}^{+}_{\mathrm{s},k}\}$ using the innovation-based approach

$$\text{Control input} \qquad \boldsymbol{u}_{\mathrm{s},k} = \mathrm{diag}\Big\{\mathbf{P}^{yy}_{k} - \hat{\mathbf{R}}_{k-1}\Big\} \tag{2.47g}$$

$$\text{Measured output} \qquad \boldsymbol{y}_{\mathrm{s},k} = \mathrm{diag}\{\hat{\mathbf{P}}^{yy}_{k}\} \quad \Rightarrow \text{cf. Eq. (2.27)} \tag{2.47h}$$

$$\text{Output matrix} \qquad \mathbf{C}_{\mathrm{s},k} = \mathbf{I} \tag{2.47i}$$

Opt. 2B: Measurement noise estimation $\hat{\mathbf{R}}_k = \mathrm{diag}\{\hat{\boldsymbol{x}}^{+}_{\mathrm{s},k}\}$ using the residual-based approach

$$\text{Control input} \qquad \boldsymbol{u}_{\mathrm{s},k} = \mathrm{diag}\Big\{\boldsymbol{\mathcal{K}}_k\Big(\mathbf{P}^{yy}_{k} - \hat{\mathbf{R}}_{k-1}\Big)\boldsymbol{\mathcal{K}}^{\mathrm{T}}_{k}\Big\} \quad \Rightarrow \text{cf. Eq. (2.40)} \tag{2.47j}$$

$$\text{Measured output} \qquad \boldsymbol{y}_{\mathrm{s},k} = \mathrm{diag}\{\hat{\widetilde{\mathbf{P}}}_k\} \quad \Rightarrow \text{cf. Eq. (2.33)} \tag{2.47k}$$

$$\text{Output matrix} \qquad \mathbf{C}_{\mathrm{s},k} = \boldsymbol{\mathcal{K}}_k \circ \boldsymbol{\mathcal{K}}_k \tag{2.47l}$$

Finally, App. A.4 (p. 146) provides a summary of a dedicated slave CKF so that master and slave filter can be realized as a single filter class (obviously parametrized with different designs and system models, inputs and outputs).

2.5.3 Maximum-Likelihood Estimation

As the second adaptation rule, the maximum likelihood estimation is introduced as alternative to the MS filter. It has been originally proposed by Mehra [131, 130] and was later refined by Maybeck [127]. In the following, the approach by Maybeck is briefly outlined.

From the rule of Bayes, we know that the a posteriori joint probability density of the state $\boldsymbol{x}_k$ and the unknown, but constant parameters $\boldsymbol{\vartheta}$ can be written as

$$p(\boldsymbol{x}_k,\boldsymbol{\vartheta}|\mathbf{Y}_k) = p(\boldsymbol{x}_k|\boldsymbol{\vartheta},\mathbf{Y}_k)\, p(\boldsymbol{\vartheta}|\mathbf{Y}_k) \tag{2.48}$$

where $\mathbf{Y}_k = \left\{\boldsymbol{y}_0\, \boldsymbol{y}_1\, \ldots\, \boldsymbol{y}_{k-1}\, \boldsymbol{y}_k\right\}$ entails all the past measurements and wherein $p(\boldsymbol{\vartheta}|\mathbf{Y}_k)$ is denoted as the marginal probability density of $\boldsymbol{\vartheta}$ given $\mathbf{Y}_k$. According to Bayes' theorem, this density is rewritten as

$$p(\boldsymbol{\vartheta}|\mathbf{Y}_k) = \frac{p(\mathbf{Y}_k|\boldsymbol{\vartheta})\, p(\boldsymbol{\vartheta})}{p(\mathbf{Y}_k)} \tag{2.49}$$

wherein $p(\boldsymbol{\vartheta})$ is the a priori distribution of the parameters (if given/available, otherwise dropped). $p(\mathbf{Y}_k)$ is a constant and independent of $\boldsymbol{x}_k$ and $\boldsymbol{\vartheta}$. The conditional density function $p(\mathbf{Y}_k|\boldsymbol{\vartheta})$ can be perceived as a product of its prior distributions, having the property of transition functions [203] (p. 322), such that

$$p(\mathbf{Y}_k|\boldsymbol{\vartheta}) = \prod_{i=1}^{k} p(\boldsymbol{y}_i|\boldsymbol{\vartheta}, \mathbf{Y}_{i-1}) \tag{2.50}$$

holds. Then, Eq. (2.48) can be restated, using Eqs. (2.49) and (2.50), as

$$p(\boldsymbol{x}_k,\boldsymbol{\vartheta}|\mathbf{Y}_k) = \frac{p(\boldsymbol{x}_k|\boldsymbol{\vartheta},\mathbf{Y}_k)\, p(\boldsymbol{\vartheta})}{p(\mathbf{Y}_k)} \prod_{i=1}^{k} p(\boldsymbol{y}_i|\boldsymbol{\vartheta}, \mathbf{Y}_{i-1}) \tag{2.51}$$

where hereafter the two conditional densities are assumed to be Gaussian, such that

$$p(\boldsymbol{x}_k|\boldsymbol{\vartheta},\mathbf{Y}_k) = \frac{(2\pi)^{-n_x/2}}{|\mathbf{P}_k^+(\boldsymbol{\vartheta})|^{\frac{1}{2}}} \exp\left\{-\frac{1}{2}\Big(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k^+(\boldsymbol{\vartheta})\Big)^{\mathrm{T}}\Big(\mathbf{P}_k^+(\boldsymbol{\vartheta})\Big)^{-1}\Big(\boldsymbol{x}_k - \hat{\boldsymbol{x}}_k^+(\boldsymbol{\vartheta})\Big)\right\} \tag{2.52}$$

$$p(\boldsymbol{y}_i|\boldsymbol{\vartheta},\mathbf{Y}_{i-1}) = \frac{(2\pi)^{-n_y/2}}{|\mathbf{P}_i^{yy}(\boldsymbol{\vartheta})|^{\frac{1}{2}}} \exp\left\{-\frac{1}{2}\Big(\boldsymbol{y}_i - \hat{\boldsymbol{y}}_i(\boldsymbol{\vartheta})\Big)^{\mathrm{T}}\Big(\mathbf{P}_i^{yy}(\boldsymbol{\vartheta})\Big)^{-1}\Big(\boldsymbol{y}_i - \hat{\boldsymbol{y}}_i(\boldsymbol{\vartheta})\Big)\right\} \tag{2.53}$$

holds. Therein, $|*|$ indicates the matrix determinant. These equations show several familiar quantities (provided by the Kalman filter) such as the a posteriori mean and covariance estimates as well as the predicted measurement and the innovation covariance. Note that these depend explicitly on the model/filter parameters $\boldsymbol{\vartheta}$.

The underlying idea of the MLE is to maximize the posterior probability density $p(\boldsymbol{x}_k,\boldsymbol{\vartheta}|\mathbf{Y}_k)$, defined by Eq. (2.51), over a given time interval. For this epoch, only the last N measurements are considered in the optimization problem so that $\mathbf{Y}_k$ is readily replaced by $\mathbf{Y}_{k|N} = \left\{\boldsymbol{y}_{k-N+1}\,\boldsymbol{y}_{k-N+2}\,\cdots\,\boldsymbol{y}_{k-1}\,\boldsymbol{y}_k\right\}$. Moreover, it is assumed that the parameters $\boldsymbol{\vartheta}$ are constant (at least) over any given interval of N samples which is usually justified as the best a priori assumption [128] (p. 73).

As an objective function, the so-called log-likelihood function L is defined as follows:

$$L(\boldsymbol{x}_k,\boldsymbol{\vartheta}) = \ln\left\{p(\boldsymbol{x}_k,\boldsymbol{\vartheta}\,|\mathbf{Y}_{k|N})\right\}\,. \tag{2.54}$$

The reason is that, due to the monotony property of the logarithm, the location of the optimum $\boldsymbol{x}_k^*,\boldsymbol{\vartheta}^*$ is preserved, but Eq. (2.54) is much easier to evaluate analytically than Eq. (2.51) would be.[12]

Long story short, the optimum is found by setting the derivative of the likelihood function to zero which (after several assumptions and simplifications) yields the so-called pseudo likelihood function

$$\frac{\partial L(\boldsymbol{x}_k,\boldsymbol{\vartheta})}{\partial\boldsymbol{\vartheta}} = \mathbf{0} \;\Rightarrow\; \sum_{i=k-N+1}^{k} \mathrm{Tr}\left\{\left[(\mathbf{P}_i^{yy})^{-1} - (\mathbf{P}_i^{yy})^{-1}\,\boldsymbol{v}_i\boldsymbol{v}_i^{\mathrm{T}}\,(\mathbf{P}_i^{yy})^{-1}\right]\frac{\partial\mathbf{P}_i^{yy}}{\partial\vartheta_j}\right\} = 0 \tag{2.55}$$

for any noise parameter ϑ_j. Now we would like to find a solution to minimize the covariance in dependence of the unknown parameters, e.g. $\mathbf{Q}_k$ and/or $\mathbf{R}_k$. Neither in the linear case nor in the nonlinear case a closed solution is found, so that a few approximations are necessary. Refer to [128] (p. 74 ff, p. 120 ff) for more details.

Consequently, Eq. (2.55) is used to obtain an estimate for the measurement noise $\mathbf{R}_k$ (provided that $\mathbf{Q}_k$ is known in advance) as

$$\hat{\mathbf{R}}_k = \frac{1}{N}\left[\sum_{i=k-N+1}^{k}\boldsymbol{v}_i\boldsymbol{v}_i^{\mathrm{T}}\right] - \mathbf{P}_k^{yy} + \hat{\mathbf{R}}_{k-1} \tag{2.56}$$

assuming an ergodic process and hence the sample averages are replaced by time averages. This relation is known as the so-called Maybeck estimator for measurement noise [127]. In Eq. (2.56), the first term on the right-hand side is recognised as the innovation sample covariance matrix from Eq. (2.27). Moreover, a solution for $\mathbf{Q}_k$ (given $\mathbf{R}_k$ in advance) is obtained as follows

$$\hat{\mathbf{Q}}_k = \frac{1}{N}\left[\sum_{i=k-N+1}^{k}\widetilde{\boldsymbol{x}}_i\widetilde{\boldsymbol{x}}_i^{\mathrm{T}}\right] + \mathbf{P}_k^{+} - \left(\mathbf{P}_k^{-} - \hat{\mathbf{Q}}_{k-1}\right) \tag{2.57}$$

where, in accordance with Eq. (2.34), the first term on the right-hand side of Eq. (2.57) is once again recognised as the residual sample covariance matrix.

[12] By using the natural logarithm, the product over $p(\boldsymbol{y}_i|\boldsymbol{\vartheta},\mathbf{Y}_{i-1})$ is replaced by a sum and the exponential function in the normal distributions is eliminated which facilitates the analytical differentiation considerably.

Although the derivation has originally been based on a linear Kalman filter, the formula can also be applied directly to non-linear filters. However, the necessary simplifications to get a closed solution make the consistency of the equation doubtful [127].

Several authors have further developed the MLE or Maybeck estimator including Mohamed & Schwarz [136], Vepa [210], Oussalah & De Schutter [153], Busse et al. [36], Lee & Alfriend [116] (p.4) as well as Song et al. [200] in order to address some of the shortcomings. Refer to these publications for more details on ML-based estimation concepts.

2.6 Summary

This second chapter summarizes the relevant principles and algorithms that are required to understand and apply state estimation for wind turbines successfully.

For this purpose, the cubature Kalman filter has been introduced in greater detail, providing also an efficient square-root implementation. Moreover, an overview on adaptive filters for process and measurement noise estimation has been given. The master-slave filter and the Maybeck estimator are introduced as recommended algorithms for noise adaptation. Such an adaptation is considered as an add-on feature if no satisfactory estimation results can be obtained with an optimally designed classical SPKF. These estimation techniques pave the way to tackle the system's nonlinearity as well as to cope with time-varying, unknown noise properties effectively.

In addition, the necessary performance criteria have been put together in order to assess the estimation results in simulation and also to allow for online estimation performance management (EPM). This constitutes a necessity for industrial application in advanced control systems.

After the algorithmic basis for the present thesis has been established, the subsequent chapter will introduce the object of this research, namely the wind turbine and the physical dynamic modelling thereof.

3 Turbine Modeling and Technology

This chapter deals with the physical modeling of horizontal axis wind turbines for estimation and control purposes. This includes the three major sub-models for the wind field, the aerodynamics and the structural dynamics. Moreover, the necessary technological details are introduced. Several models are derived and proposed as internal filter models.

3.1 Introduction

Wind energy is in general a volatile, a non-controllable and an almost unpredictable resource of renewable energy. These properties are in general undesirable for every technology which shall be exploited to deliver continuously and reliably energy for private house-holds as well as industrial plants. On the other hand, the wind resource is almost everywhere and free-of-charge available, it is moreover renewable and clean which makes it a very desirable and independent resource for future energy supply. These properties are therefore very welcome. Modern WT design and control concepts must therefore integrate and incorporate the benefits and also the disadvantages to deliver predictable electrical power to the grid.

To achieve these objectives, simplified engineering models and control-oriented models are called for. These are obtained through mathematical abstraction of the real-world system by neglecting irrelevant aspects for the considered problem. Control-oriented modeling seeks for models that catch only the relevant details for the pursued application in estimation or control. The process of deriving a suitable representation based on physical laws, empirical relations and/or data driven approaches is termed mathematical modeling. Such an illustrative model is provided in Fig. 3.15 (p. 67). This approach incorporates for wind turbines at least the following three physical domains: Aerodynamics, classical mechanics and electrodynamics. The thesis brings the first two disciplines into focus while the electrical dynamics (which are comparatively faster) are either neglected or modelled by dynamic systems with small time constants. It is favourable to derive several models with increasing level of detail since the specific estimation problem (in practice) defines the suitable internal filter model.

The primary goal of this section is the derivation of internal design models for the estimators and filter algorithms, respectively. Secondly, an advanced control-oriented wind turbine model is developed for simulation purposes. The wind turbine models are divided into a wind field sub-model, an aerodynamic sub-model and an elastodynamic sub-model. Each module may change in level of detail. The required accuracy of the model or application defines eventually its degrees of freedom and its granularity.

3.2 Wind Power Conversion

Historically, the present wind turbine designs have evolved from the ancient windmills which have been used successfully for several hundreds and thousands of years. They were mostly employed for grinding wheat and, thus, to convert mechanical energy of the wind motion into mechanical energy in the grinder. The development of plants for electrical energy generation had started approximately 150 years ago with first prototypes.

Today's strong developments towards modern large-scale wind turbines began in the late 1980's and was mainly triggered by the oil crisis at that time. Since then, the world-wide installed capacity has increased exponentially as shown in Fig. 3.1 (data from 1996 forward) which was facilitated essentially by the ongoing growth in turbine size due to technological leaps. In addition, Fig. 3.2 and Fig. 3.3 give a visual impression of this growth in rotor diameter, hub height and also installed power over the last 20 years (illustratively shown for Germany).

Although there are various types of horizontal axis and vertical axis concepts with different shapes and number of blades [72], the three-bladed horizontal axis wind turbine as the most successful rotor concept today has prevailed. No matter what concept is looked at and pursued, the development of larger turbines with monotonically growing nominal power production can be seen consistently.

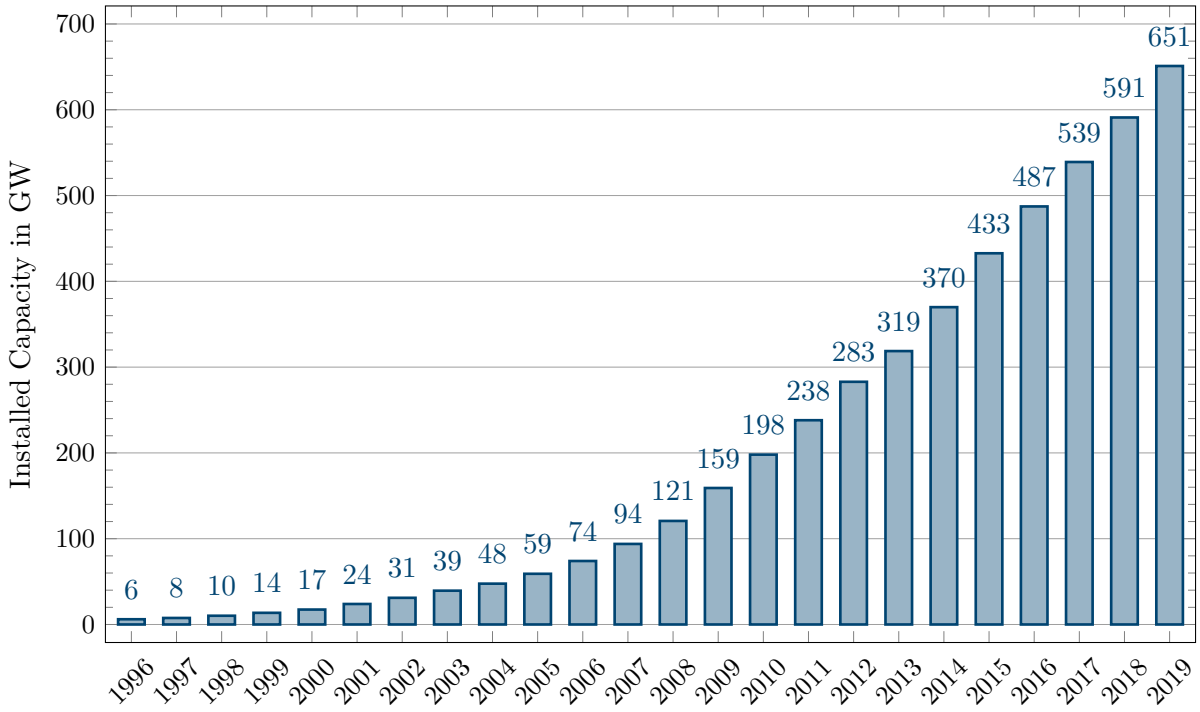

Figure 3.1: Globally installed cumulative wind energy capacity from 1996 to 2019 (own diagram based on data from the Global Wind Energy Council, GWEC)

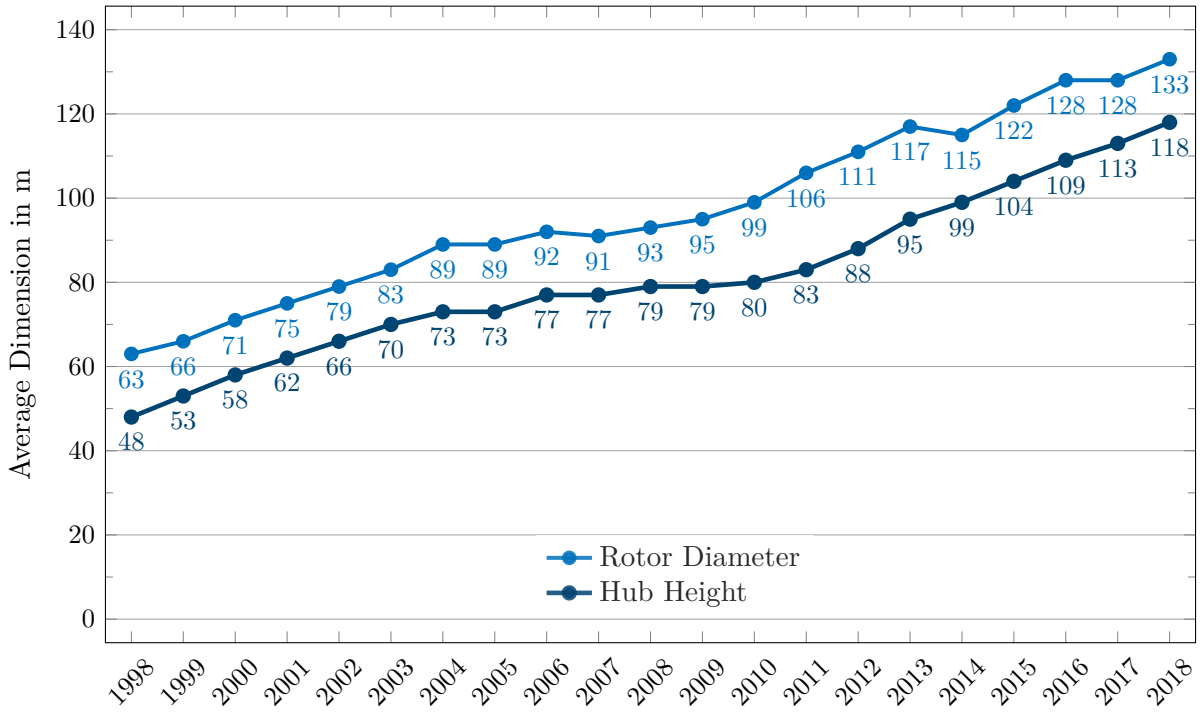

Figure 3.2: Average rotor diameters and average hub heights for newly and yearly installed onshore turbines in Germany between 1998 and 2018 (own diagram based on data from the Betreiber-Datenbasis, see http://www.windmonitor.de/[...] [accessed 2020-05-02])

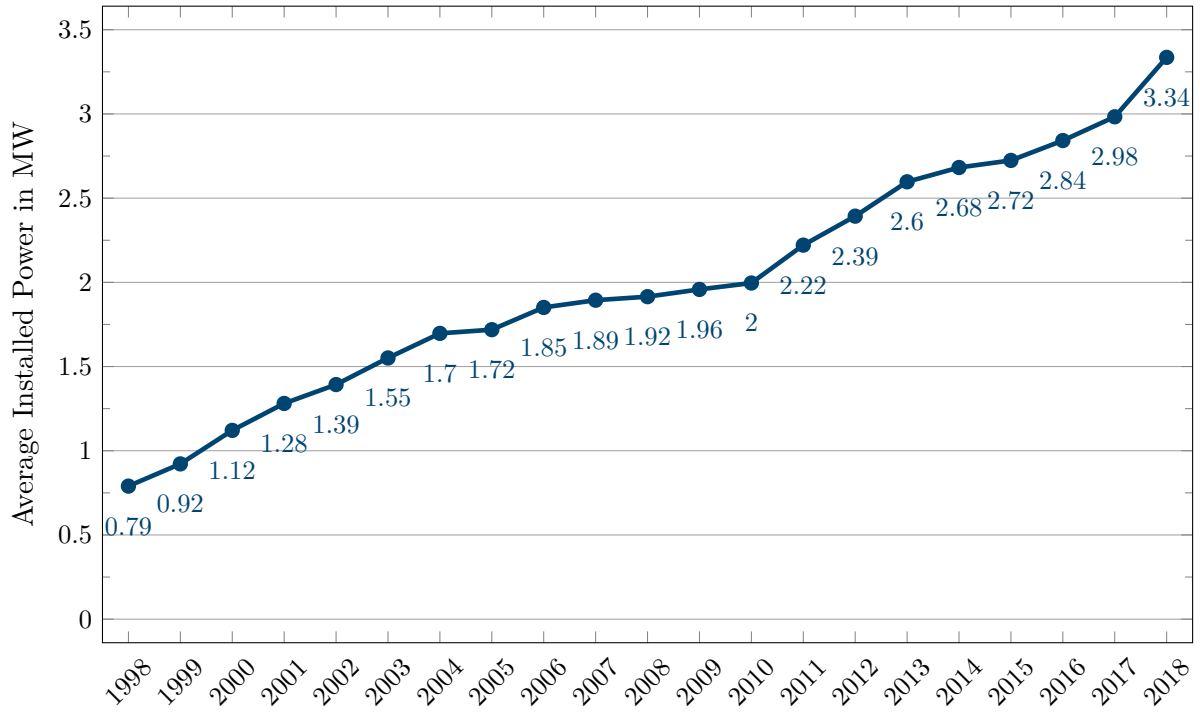

Figure 3.3: Average installed power for newly and yearly installed onshore turbines in Germany between 1998 and 2018 (own diagram based on data from the Betreiber-Datenbasis in Germany, see also http://www.btrdb.de [accessed 2020-05-02])

3.2.1 Actuator Disk Theory and the Betz Limit

The main reason for developing bigger and larger power plants is the overarching goal to reduce the LCoE. With larger turbines the AEP is increased considerably while the costs at the same are expected to grow less and therefore not in the same ratio. To boost the nominal power output of a single plant practically, one must design and build larger rotors with longer blades which cover a larger rotor-swept area $A = \pi R^2$ and turbines with larger hub heights H (cf. Fig. 3.2). This can be seen directly from the mechanical power equation

$$P_{\text{avail}} = \frac{\varrho}{2} \pi R^2 v_\infty^3 \tag{3.1}$$

where P_{avail} is also denoted as the available wind power. Hence, the available power depends cubically on the free-stream velocity v_∞ (which is a function of H), linearly on the air mass density ϱ and, essentially, increases with square of the blade tip radius R.

It is intuitively understandable that the power available in Eq. (3.1) cannot be harvested completely [15]. That is why the air must preserve a fraction of its kinetic energy to flow entirely through the rotor plane. The German physicist and wind turbine pioneer Albert Betz discovered this physical limit in 1920 and derived it mathematically using the actuator disk theory [14]. For this purpose, Betz introduced the so-called power coefficient C_{P}.[13] In accordance with [34, 70], the power coefficient is defined as the ratio of aerodynamic rotor power to available wind power

$$C_{\text{P}} \equiv \frac{P_{\text{a}}}{P_{\text{avail}}} = 4\,a\,(1-a)^2 \tag{3.2}$$

where $a \in [0, 1]$ is the so-called axial induction factor. The latter defines the wind speed reduction (the axial induction) in the rotor plane by

$$v_{\text{R}} = (1-a)\,v_\infty \tag{3.3}$$

and therefore as a relation to the free-stream velocity v_∞. The power coefficient in Eq. (3.2) has a theoretical maximum

$$C_{\text{P,max}} = \frac{16}{27} \quad \text{obtained for} \quad a = \frac{1}{3} \tag{3.4}$$

which is mostly referred to as Betz limit although Lanchester-Betz-Joukowsky limit would be historically more appropriate [148, 113, 14, 98]. Thus, in the ideal case it is possible to harvest 59.3 % of the wind's kinetic energy by reducing the wind speed in the rotor plane to one third of the free-stream velocity. This theoretical maximum is a physical limit. It is a very remarkable result since Betz's theory is valid for every thinkable wind energy

[13] In the original publication [14], Betz used $C_l = \eta P_{\text{a}}/P_{\text{avail}}$ as definition for the aerodynamic power coefficient (in German *Leistungsziffer*) where he considered the mechanical efficiency η as part of his equation. This is contrary to Eq. (3.2) which is today used in the majority of publications (cf. [34, 70]). Under the assumption $\eta = 1$, both approaches are obviously equivalent.

conversion system and the limit was derived completely independent of a specific actuator design or concept.

Apart from the power coefficient C_P, the aerodynamic coefficient for rotor thrust C_T represents a dimensionless force (interacting with the wind stream) defined by

$$C_\mathrm{T} \equiv \frac{F_\mathrm{T}}{F_\mathrm{ref}} = 4\,a\,(1-a) \tag{3.5}$$

where F_T is the rotor thrust force and its reference value is

$$F_\mathrm{ref} = \frac{\varrho}{2}\pi R^2 v_\infty^2\,. \tag{3.6}$$

Hence, rotor thrust and power are always linked to each other by only one independent variable, the axial induction factor a. As a consequence, mechanical power can never be extracted from the wind without generating rotor thrust that must be absorbed by the rotor blades, the tower and its foundation, respectively. For the optimal a in Eq. (3.4) the thrust coefficient C_T is 8/9.

3.2.2 Blade Element Momentum Theory

Due to (energy) losses of the non-ideal actuator the maximum $C_\mathrm{P,max}$ cannot be achieved in reality. The main reasons are the limitations of the real rotor compared to the theoretical disc actuator which are the finite number of blades, the wake rotation (tangential induction), the blade tip losses and the rotor hub losses, and the drag/friction losses around the aerodynamic blade profiles. In total, these accumulated effects lower the power coefficient substantially if this rotor is not carefully designed. The resulting C_P-curves of the real rotor still exhibit one optimum value (cf. Fig. 3.4) for the power coefficient

$$C_\mathrm{P}^* \leq C_\mathrm{P,max}\ . \tag{3.7}$$

In order to distinguish both values, the theoretical maximum is denoted as $C_\mathrm{P,max}$ and the (practical) optimum power coefficient as C_P^*. The latter one is still optimal however only for the chosen design. Typical values for modern WTs range inbetween 0.4 and 0.5 under advantageous operation conditions (ensured by control).

To compute the aerodynamic coefficients for an arbitrary rotor design, the established approach is to use the Blade Element Momentum (BEM) theory with some corrections [138, 70]. This method combines the momentum theory of Betz (in German *Impulstheorie*) with blade-element theory (in German *Profiltheorie*) applied to the profile sections along the blade span. In principle the blade is divided into several discrete rotor annuli with a typical cross-section shown in Fig. 3.5.

First, the inflow angle ϕ_i for the i-th blade section is defined according to

$$\tan\phi_i = \frac{1-a_i}{1+a_i'}\,\frac{v_\infty}{\Omega r_i} = \frac{1-a_i}{1+a_i'}\,\frac{R}{r_i}\lambda^{-1} \tag{3.8}$$

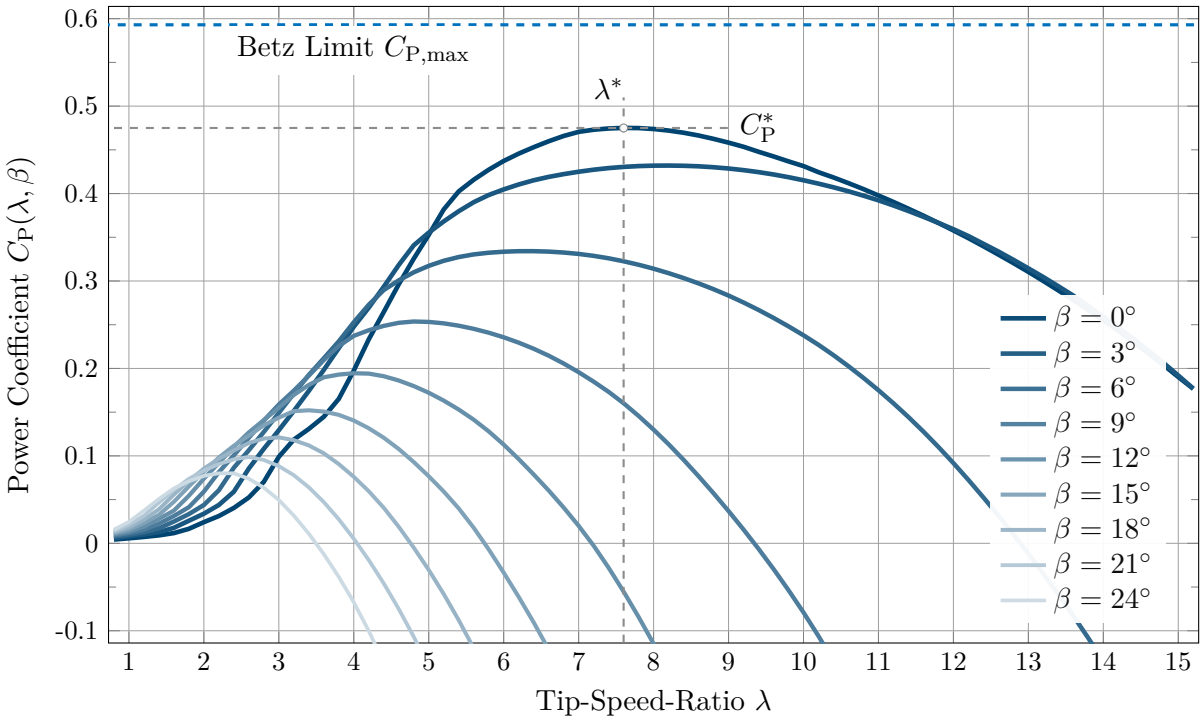

Figure 3.4: Aerodynamic power coefficient C_P as a function of the tip-speed-ratio λ and parametrized by the pitch angle β including the optimal design point $C_P^* = C_P(\lambda^*)$ (illustratively depicted for the NREL 5 MW reference turbine)

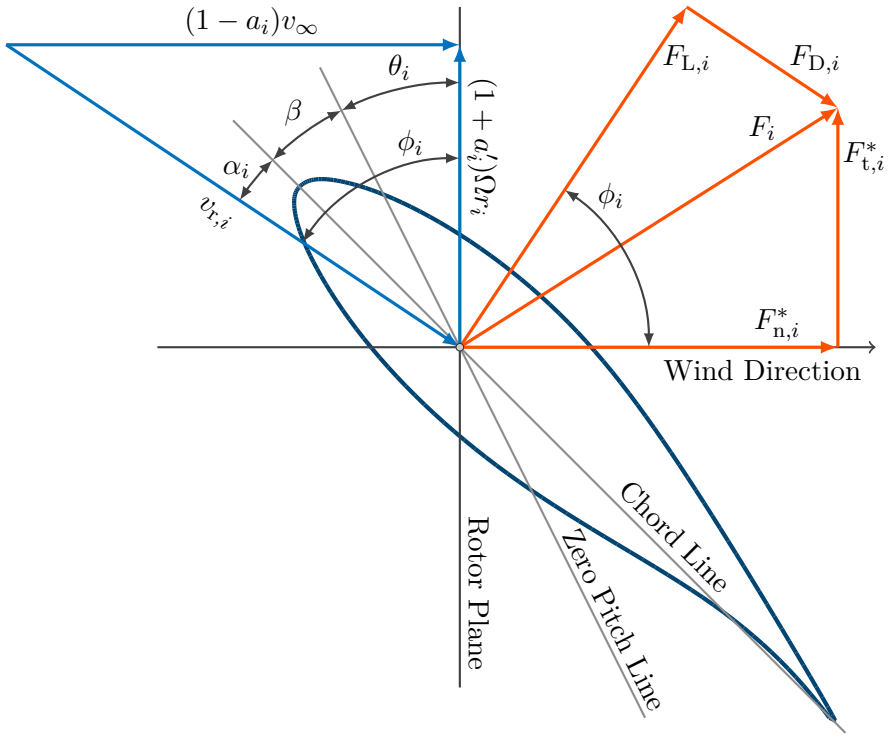

Figure 3.5: Velocities and aerodynamic forces at a cross-section of an illustrative rotor blade profile assuming that the yaw angle is zero and thus no yaw misalignment (the zero pitch line and the chord line are identical for $\beta = 0$, the illustrative aerofoil is NACA64-A17)

wherein a_i is the axial induction factor and a'_i is the tangential induction factor. r_i is the local radius of the blade section and R the blade tip radius. The so-called tip-speed-ratio (or TSR for short)

$$\lambda(t) = \frac{R\,\Omega(t)}{v_\infty(t)} \tag{3.9}$$

is one of the most important time-dependent variables for wind turbine design and operation. The design parameter λ^* is the optimal TSR for $C_\mathrm{P}^* = C_\mathrm{P}(\lambda^*)$. λ is conceivable as a dimensionless rotor speed.

The relative inflow velocity $v_{\mathrm{r},i}$ at each element can be obtained geometrically as

$$v_{\mathrm{r},i} = \sqrt{(1-a_i)^2 v_\infty^2 + (1+a'_i)^2 \Omega^2 r_i^2} \tag{3.10}$$

which can be understood from Fig. 3.5. The lift and drag forces at the section are defined by the lift and drag coefficients as well as section geometry according to

$$F_{\mathrm{L},i}(\alpha_i, v_{\mathrm{r},i}) = \frac{\varrho}{2} c_i b_i C_{\mathrm{L},i}(\alpha_i) v_{\mathrm{r},i}^2 \tag{3.11}$$

$$F_{\mathrm{D},i}(\alpha_i, v_{\mathrm{r},i}) = \frac{\varrho}{2} c_i b_i C_{\mathrm{D},i}(\alpha_i) v_{\mathrm{r},i}^2 \tag{3.12}$$

where $F_{\mathrm{L},i}$ is perpendicular to $v_{\mathrm{r},i}$ and $F_{\mathrm{D},i}$ is in direction of $v_{\mathrm{r},i}$. $C_{\mathrm{L},i}$ and $C_{\mathrm{D},i}$ are the so-called lift and drag coefficients which depends only on the angle of attack α_i (cf. Fig. 3.5). Additionally, Fig. 3.6 illustrates the curves for these coefficients for two selected aerofoils.

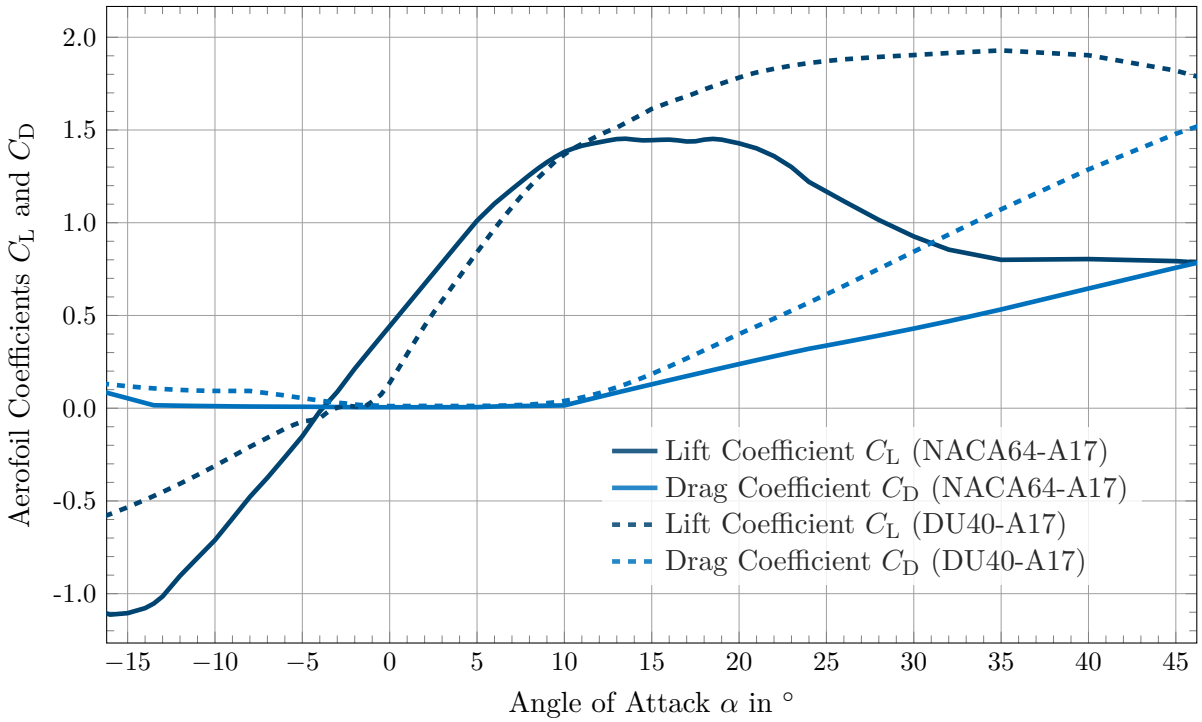

Figure 3.6: Aerodynamic lift and drag coefficients as a function of the angle of attack α in the relevant area of interest (illustratively depicted for the aerofoils NACA64-A17 and DU40-A17)

The resulting aerodynamic forces at the section i is F_i. Note that Tab. 3.1 provides the descriptions of the relevant parameters and variables for the BEM computation. The BEM theory also provides semi-analytical expressions for the axial and tangential induction factor in each blade section i as follows

$$a_i = \left(\frac{4 \sin^2 \phi_i}{\sigma_i \, C_{\mathrm{n},i}(\phi_i)} + 1 \right)^{-1} \tag{3.13a}$$

$$a'_i = \left(\frac{4 \sin \phi_i \cos \phi_i}{\sigma_i \, C_{\mathrm{t},i}(\phi_i)} - 1 \right)^{-1} \tag{3.13b}$$

where the normal and tangential force coefficients are

$$C_{\mathrm{n},i}(\phi_i, \beta) = \cos \phi_i \, C_{\mathrm{L},i}\Big(\phi_i - \theta_i - \beta\Big) + \sin \phi_i \, C_{\mathrm{D},i}\Big(\phi_i - \theta_i - \beta\Big) \tag{3.14a}$$

$$C_{\mathrm{t},i}(\phi_i, \beta) = \sin \phi_i \;\, C_{\mathrm{L},i}\Big(\phi_i - \theta_i - \beta\Big) - \cos \phi_i \, C_{\mathrm{D},i}\Big(\phi_i - \theta_i - \beta\Big) \tag{3.14b}$$

where the local angle of attack constitutes

$$\alpha_i = \phi_i - \theta_i - \beta \tag{3.15}$$

and the blade's local solidity for a rotor with n_{bl} individual blades is

$$\sigma_i = \frac{n_{\mathrm{bl}} \, c_i}{2 \pi r_i} \; . \tag{3.16}$$

Finally, the normal and tangential aerodynamic forces at each section (cf. Fig. 3.5) write

$$F^*_{\mathrm{n},i}(\phi_i, \beta, v_{\mathrm{r},i}) = \frac{\varrho}{2} c_i b_i C_{\mathrm{n},i}(\phi_i, \beta) v^2_{\mathrm{r},i} \tag{3.17}$$

$$F^*_{\mathrm{t},i}(\phi_i, \beta, v_{\mathrm{r},i}) = \frac{\varrho}{2} c_i b_i C_{\mathrm{t},i}(\phi_i, \beta) v^2_{\mathrm{r},i} \tag{3.18}$$

which are needed to formulate the aerodynamic forces perpendicular to the rotor plane and also in-plane. More details on the derivation of the above equations are provided, for instance, by Hansen [70].

Analysing the Eqs. (3.8), (3.13) and (3.14) together, reveals that there is actually only one independent variable ϕ_i and two freely-selectable parameters (the tip-speed-ratio λ and the blade pitch angle β) which are fixed for each single optimization. Hence, solving the BEM for chosen pairs of λ and β, and solving consequently Eq. (3.8), can be comprehended as minimizing a cost function

$$J\Big(\phi_i(\lambda, \beta)\Big) = \frac{1}{2} \left(\tan \phi_i - \frac{1 - a_i(\phi_i)}{1 + a'_i(\phi_i)} \, \frac{R}{r_i} \lambda^{-1} \right)^2 \tag{3.19}$$

which is an implicit nonlinear equation of the inflow angle ϕ_i by nature. Thus, the minimum

$$\hat{\phi}_i(\lambda, \beta) = \operatorname{argmin} \Big\{ J\Big(\phi_i(\lambda, \beta)\Big) \Big\} \tag{3.20}$$

is commonly obtained using numerical methods (e.g. Newton-Raphson methods [81]). Practically to get more accurate results solving the optimization problem Eq. (3.20), the Glauert's

Table 3.1: Variables and model parameters for numerical BEM computation

Variable	Description
α, α_i	(local) angle of attack which defines mainly lift and drag coefficients
ϕ, ϕ_i	(local) inflow angle at the aero-profile between wind and lift force direction
β	the blade pitch angle which is a control actuating variable
θ_i	local twist angle (defined by blade design)
λ	tip-speed-ratio
$\Omega, \dot{\varphi}$	rotor angular speed which is actively regulated by WT controller
σ_i	local solidity of the blade element (fraction of the annular rotor area)
a, a_i	(local) axial induction factor
a'_i	local tangential induction factor
c_i	local chord length
r_i	local blade element radius
b_i	local section width
$v_{\mathrm{r},i}$	local relative flow velocity at the blade profile
$C_{\mathrm{L},i}$	local lift force coefficient of section profile (only dependent on α_i)
$C_{\mathrm{D},i}$	drag force coefficient of section profile (only dependent on α_i)
$C_{\mathrm{n},i}$	normal force coefficient for blade element (function of α_i and β)
$C_{\mathrm{t},i}$	tangential force coefficient for blade element (function of α_i and β)

correction and Prandtl's tip and hub loss correction factors need to be considered as well [68, 70, 158].[14]

With Eq. (3.20), the axial induction $\hat{a}_i(\lambda,\beta)$ and the tangential induction $\hat{a}'_i(\lambda,\beta)$, as well as the force coefficients $\widehat{C}_{\mathrm{t},i}(\lambda,\beta)$ and $\widehat{C}_{\mathrm{n},i}(\lambda,\beta)$ for each blade element i are determined using Eqs. (3.13) and (3.14). Finally linking the above derivations, the aerodynamic thrust and power coefficients of the rotor are easily obtained numerically using

$$C_{\mathrm{P}}(\lambda,\beta) = n_{\mathrm{bl}} \sum_{i=1}^{n_{\mathrm{sec}}} \left(\Big(1-\hat{a}_i(\lambda,\beta)\Big)^2 + \lambda^2 \Big(1+\hat{a}'_i(\lambda,\beta)\Big)^2 \left(\frac{r_i}{R}\right)^2 \right) \widehat{C}_{\mathrm{t},i}(\lambda,\beta) \frac{c_i\, b_i}{A}\, \frac{r_i}{R}\, \lambda \tag{3.21a}$$

$$C_{\mathrm{T}}(\lambda,\beta) = n_{\mathrm{bl}} \sum_{i=1}^{n_{\mathrm{sec}}} \left(\Big(1-\hat{a}_i(\lambda,\beta)\Big)^2 + \lambda^2 \Big(1+\hat{a}'_i(\lambda,\beta)\Big)^2 \left(\frac{r_i}{R}\right)^2 \right) \widehat{C}_{\mathrm{n},i}(\lambda,\beta) \frac{c_i\, b_i}{A} \tag{3.21b}$$

where n_{sec} is the number of blade elements, c_i is the element chord length and b_i is the element width. The optimal power coefficient for a given rotor design is then determined by

$$C_{\mathrm{P}}^{*} = \max\Big\{ C_{\mathrm{P}}(\lambda,\beta) \Big\}\,. \tag{3.22}$$

To make a long story short: The BEM theory facilitates the derivation of engineering aerodynamic models based on aerodynamic power C_{P} and rotor thrust C_{T} only as function of TSR λ and blade pitch angle β. It is a relevant part of the control-oriented modeling for wind

[14] A different approach is to run the aerodynamic simulator in a gridding mode (controlling the TSR with fixed pitch angle) and thus obtain the solution in a controlled simulation environment without optimization.

turbines. Fig. 3.7 introduces the characteristic maps illustratively for aerodynamic power $C_{\mathrm{P}}(\lambda, \beta)$, aerodynamic torque $C_{\mathrm{M}}(\lambda, \beta) = \lambda^{-1} C_{\mathrm{P}}(\lambda, \beta)$ and aerodynamic thrust $C_{\mathrm{T}}(\lambda, \beta)$ which have been evaluated for the 5 MW reference turbine using the above approach. These illustrations also include the nominal operating curve that is enforced by the WTC.

In the following, the focus is laid on the three-bladed horizontal axis wind turbines (HAWT) since they are the by far leading industrial concept for multi-MW turbines today. Yet, the ideas and methods of state estimation may be analogously applied to other types like vertical axis wind turbines (VAWT).

3.3 Turbine Technology

Modern three-bladed HAWTs are composed of several structural components to fulfil the wind turbine's single purpose to extract energy from the wind and convert it to electrical energy. This section describes first the necessary main components and also the used sensor equipment. After that, a brief overview of wind turbine control is given. Finally, a glimpse on current developments in wind turbine application is provided.

3.3.1 Main Components and Control Actuators

The control-oriented model must consider the system dynamics of the main components that are relevant to fulfil the control and load reduction objectives. Fig. 3.8 shows a typical modern wind turbine and the plant's main components (next to each other).

The foundation (or platform for floating concepts) absorbs the aerodynamic and mechanical loads acting on the tower and the rotor, respectively. It must guide them safely into the ground and must avoid collapsing of the entire *inverse pendulum like* structure of the WT.

The tower supports the nacelle with the complete drive-train and provides the necessary height for the rotor. Usually it is steel and/or concrete design, though a first turbine with 1.5 MW rating was reported recently where a completely wooden tower design was put into practice[15]. This tower design is said to be superior to the common steel construction due to the intrinsically higher material damping coefficients.

The tower poses aerodynamically a resistance for the wind stream and thus has to bear additional drag forces. Moreover, the resistance causes wind speed reductions for the blades which is known as the tower dam effect. It causes significant cyclic loads on the rotor, the drive-train and the blades. Often the tower houses the grid connection at the bottom.

The nacelle houses the generator, its power electronics, the controller, the gear-box and rotor bearings. Moreover, the nacelle is pivoted around the vertical axis to allow for adjustments

[15] `https://newatlas.com/timbertower-wooden-wind-turbine/25007/` [accessed 2020-01-14]

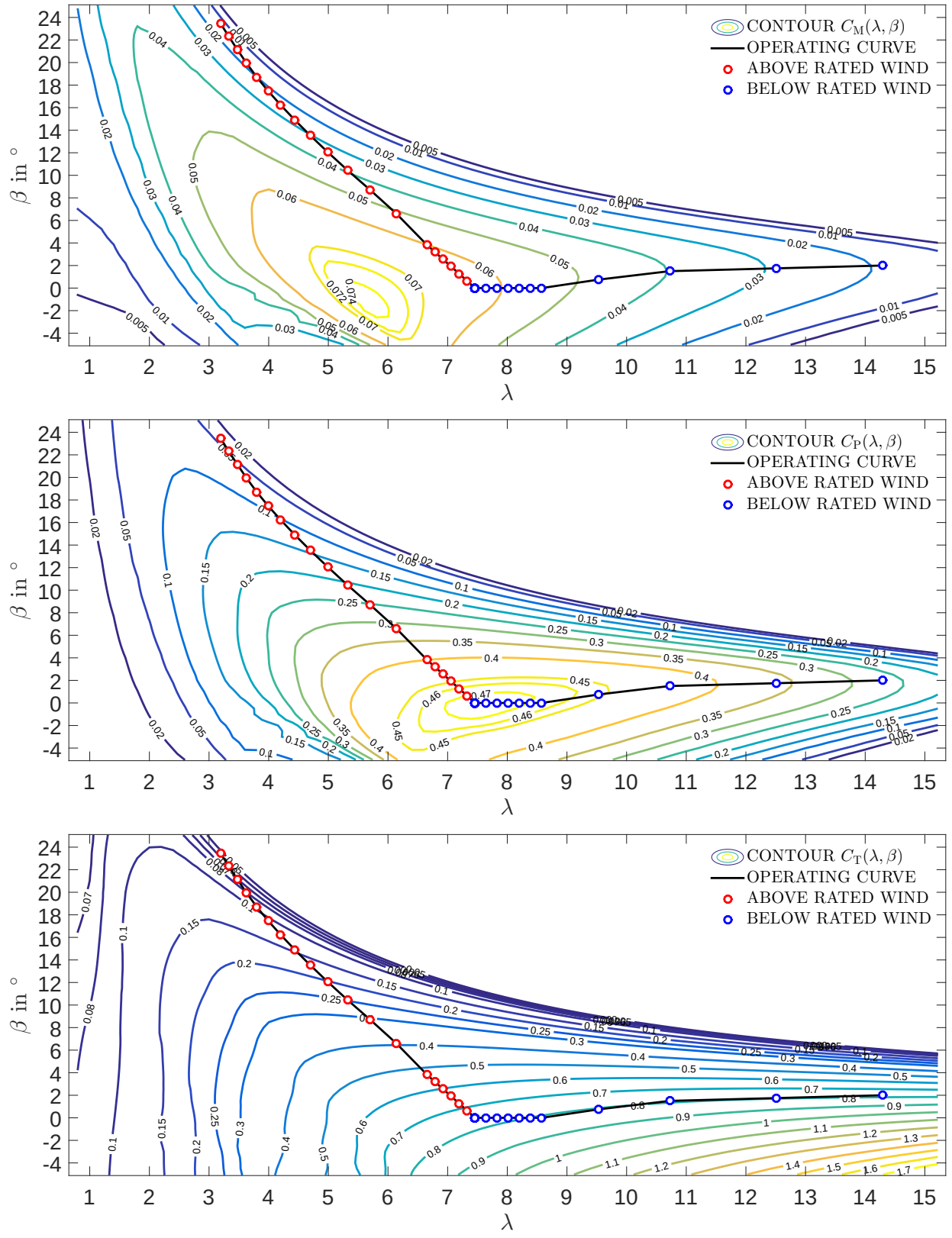

Figure 3.7: Aerodynamic coefficients for rotor torque $C_M(\lambda,\beta)$, power $C_P(\lambda,\beta)$ and thrust $C_T(\lambda,\beta)$ of the NREL 5 MW reference turbine [169]. The optimum power coefficient $C_P^* = 0.475$ is drawn from the map in the middle and is reached at an optimum tip-speed-ratio $\lambda^* = 7.6$

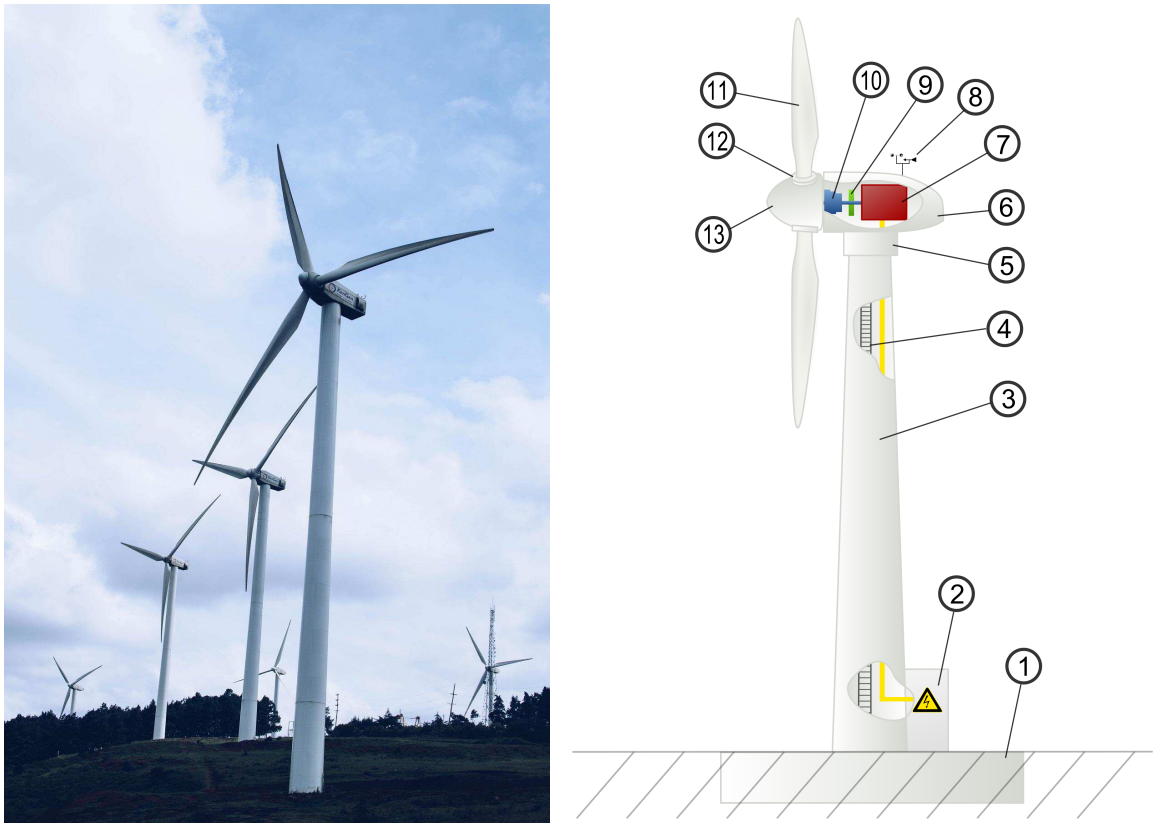

Figure 3.8: Three-bladed wind turbines in a small wind farm within complex and hilly terrain (left-hand side) and sketch of the wind turbine main components (right-hand side): (1) Foundation, (2) connection to electric grid, (3) tower, (4) access ladder, (5) wind orientation control/yaw adjustment, (6) nacelle housing, (7) electrical generator, (8) wind anemometer and wind vane, (9) mechanical brake, (10) transmission gearbox, (11) rotor blade, (12) blade pitch control/actuator, (13) rotor hub, Source: https://commons.wikimedia.org/[...] [accessed 2020-01-14]

due to changing wind directions. The yaw drive with its roller bearings act as connection to the tower top which is used to adjust the rotor always optimally to the wind direction or to reach a safe position during storms and emergency stops.

The power electronics and the generator are placed in the nacelle. They must ensure safe autonomous operation since the wind turbine operator only defines reference values which the power production controller tries to realize. Redundancy must be implemented in hardware and software to be able to manage failures e.g. in blade pitch control which results in increased loads or worse.

The drive-train usually consists of a generator, the multi-stage helical gear-box, the rotor and safety couplings. With the so-called Danish concept, the asynchronous generator is directly coupled to the grid which has some major drawbacks [34] (p. 2-6). Therefore, today mostly the doubly-fed asynchronous generator is used which allows a variable speed operation and requires only a partial inverter to feed the rotor circuit. Since the gearbox is subject to increased wear due to fluctuating wind loads, also the gear-less direct-drive concepts with

the slowly rotating permanent magnet synchronous generator (with high pole count) with full inverter constitutes an competitive alternative to the DFIG with partial inverter.

The rotor shaft is connected to the drive-train via a brake system and is also attached to the hub on the other side which accommodates the three rotor blades and its independent pitch drives. These allow to correct the individual blade pitch angles in high winds to limit the aerodynamic power and to reduce loads. Moreover, today's turbines must be able to realize an emergence shut-down with only two blades, cf. [34] (p. 518f).

Finally, the largest and most important component is the WT rotor with its three blades itself. The aerodynamic design must be optimized for optimum C_P^* (cf. Sect. 3.2.2) while the mechanical design is enforced to be light weight, low-mass and have preferably a large rotor diameter. Since there are mechanical limitations for constructing larger blades due to available materials, active load alleviation strategies are often propagated for recent designs [23]. These strategies require reliable sensors and actuators in order to allow a safe operation for more than twenty years. Refer to Sect. 3.3.3 for more information on wind turbine control.

The main actuators for control purposes are the electrical generator, the three blade pitch drives and the yaw drive. Due to the huge rotor inertia and increased wear resulting from excessive yaw activity, the last actuator is not applicable for control in 100 ms cycles (but rather in cycles of several minutes and is therefore rather slow acting), cf. [34] (p. 477/481ff). Therefore, it is not considered in the following.

3.3.2 Measurement Instrumentation

Modern wind turbines are equipped with several different sensors. These provide the relevant information in order to supervise and control the current state of operation and the individual components.

The plant's instrumentation includes multiple sensors for generator and rotor speed, yaw-rate as well as wind speed and wind direction [34] (p. 475 f). Moreover, there are multiple others several temperature sensors for oil cycles, bearings and windings of the electrical generator are included. For safety reasons the electrical systems must be monitored to detect grid frequency oscillations and voltage drops, the power factor and converter faults timely.

Also diverse position sensors for accurate blade pitch angles determination, rotor azimuth and yaw angle, yaw error and wind inflow direction (wind vane) amongst others are present. Furthermore, pressure sensors for oil and cooling systems, and inertial measurement units (IMU) for axial and lateral nacelle acceleration, blade velocities and accelerations are possibly integrated.

For estimation and control purposes, most of the mentioned sensors are not directly applicable. The main sensor information for state estimation comprises of yet only a few known system inputs and system outputs. This includes

- the generator torque M_g (subset of the control inputs $\boldsymbol{u}_k$),
- the individual blade pitch angles β_b (subset of $\boldsymbol{u}_k$)[16],
- the generator speed n_g (subset of the measurement outputs $\boldsymbol{y}_k$),
- the fore-aft nacelle acceleration $\ddot{x}_\mathrm{T}$ (subset of $\boldsymbol{y}_k$),
- the side-side nacelle acceleration $\ddot{y}_\mathrm{T}$ (subset of $\boldsymbol{y}_k$), and
- the rotor azimuth angle φ (subset of $\boldsymbol{y}_k$).

All together are regarded as standard sensor configuration in modern wind turbines. The available data is evidently very important for the estimator since it contains key information about the system dynamics (to be estimated). These are exploited by the estimators in order to correct the predicted estimate $\hat{\boldsymbol{x}}_k^-$ (cf. p. 19 f).

Besides the standard sensors, light detection and ranging (or lidar for short) has emerged as promising technology for future wind turbine control [186, 185]. However, these lidar systems are still very cost-intensive and the advantage to improve the LCoE is still an open discussion. Also, more advanced sensors like blade load sensors or blade accelerometers (IMU) offer benefits and facilitate more advanced load controls like IPC [21, 196, 75, 121]. Especially sensors for blade and tower loads, drive-train and gearbox issues are installed today more frequently when load reducing control strategies are part of the design concept. Typical load sensors are optical or wired strain gauges which are potentially not as reliable as desired.

In the present thesis, the focus rests mainly on the standard sensors with exception of Sect. 5.5.2 (p. 126) because these sensors already bring along the most relevant information for the estimation tasks considered.

3.3.3 Wind Turbine Control

Modern utility-scale wind turbines have at least five independent control actuators for power production control. These are the electrical generator, the three individual blade pitch drives and the nacelle yaw drive which are used to operate the turbine preferably in optimal conditions (cf. Fig. 3.9). Therewith, the WT is eligible for variable-speed variable-pitch (VSVP) operation, cf. [72, 34], which means that the rotor speed can be adjusted for MPPT and that the pitch actuators are actively controlled to regulate the power in the full load regime.

The VSVP control strategy is preferable since it not only increases the energy capture below rated but also limits the rotor power effectively for above rated wind speeds. In addition,

[16] The actuator dynamics are not considered in the simplified models. Thus, the blade pitch angles as manipulated variables can theoretically change arbitrarily fast, knowingly that this is physically impossible for an angle or position (mechanical states).

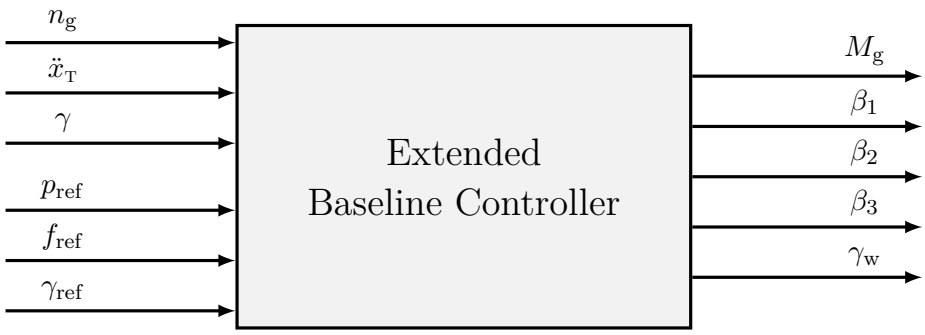

Figure 3.9: Block diagram of the extended baseline controller with measurement and reference inputs (on the left) as well as computed control actuations (on the right): $p_{ref} \in [0.5, 1]$ is the power reference, $f_{ref} \in [0, 1]$ the fatigue reference[17] and $\gamma_{ref} \in [0, 2\pi]$ the yaw reference (a separate generator speed reference is not needed necessarily)

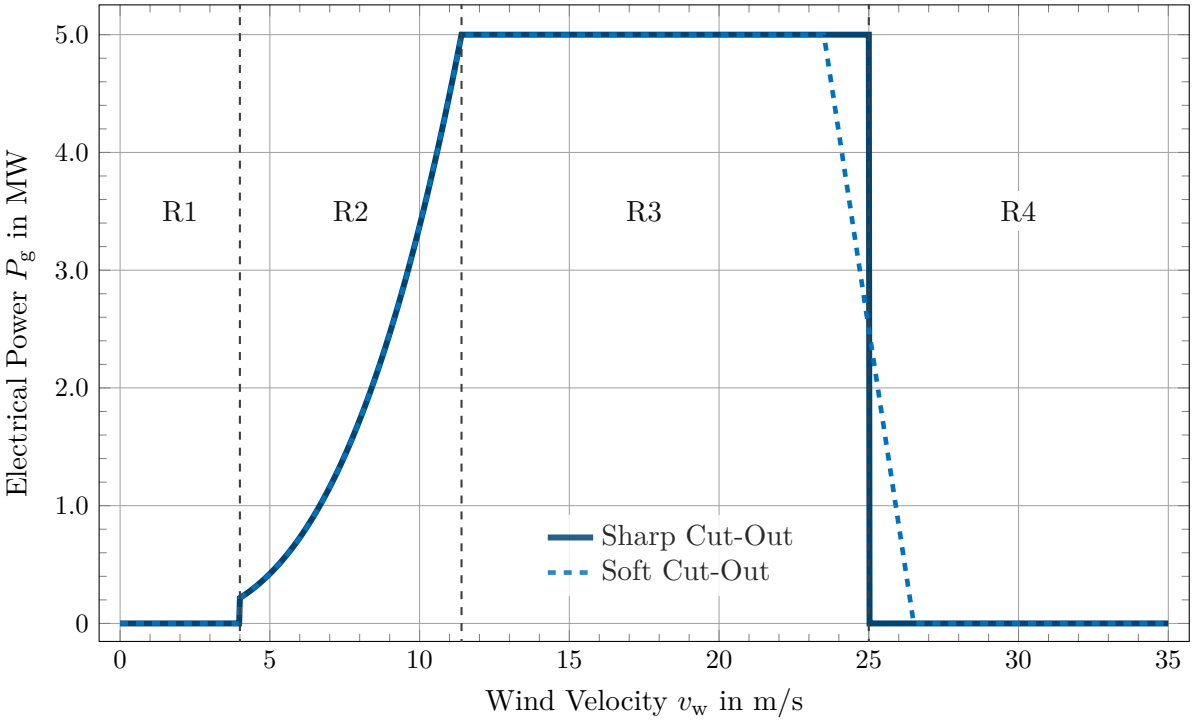

Figure 3.10: Illustrative wind turbine power curve with a nominal power of 5 MW including the operating regimes R1 to R4 where $v_{c,i} = 4\,\text{m/s}$, $v_0 = 11\,\text{m/s}$ and $v_{c,o} = 25\,\text{m/s}$ are the cut-in, nominal and cut-out wind velocities (note that the transition between R3 and R4, namely the storm management, is usually realized with a soft cut-out strategy rather than a sharp cut-out)

the mechanical fatigue loads during transition from partial load to full load operation (et vice versa) are considerably lower compared to fixed-pitch machines [17].

The typical operating regimes are displayed in Fig. 3.10. In the region R1 the contribution to the AEP is low so that the turbine cannot be operated economically here. The region R2 is also known as the partial load regime (PLR) where the available power in the wind is not enough to produce the nominal power. Therefore, the controller seeks to maximize the

power output. In contrast, region R3 is denoted as full load regime (FLR) where there is more power in the wind available than can be accommodated by the generator. In region R4 the turbine is shut-down to avoid damages due to stormy weather conditions. The regions R1 and R4 are therefore less relevant for power production control since these do not contribute to the AEP.

A wind turbine must be operated fully autonomously in all operating regimes. For this purpose, there are three main functions in every wind turbine control system, cf. Burton et al. [34] (p. 197, p. 476 ff):

1. The **supervisory control** manages and supervises the overall power production (reference values), the orderly start-up and the shut-down routines, standby, alarm management and external communication.

2. The **safety system** jumps in whenever critical operating parameters (generator speed, generator power or component vibration levels, etc.) exceed their nominal range. Thereby, the wind turbine is brought into a safe operational mode. Depending on the critical parameters and the emergency, it may or may not start independently again. The safety system intervenes only if an event or problem is serious or potentially serious. It is thus vital in any turbine.

3. The **closed-loop control system** takes responsibility for the orderly operation in changing wind conditions and wind regimes. This system is denoted as WTC in the following. As a matter of fact, it controls the above mentioned five actuators in order to achieve all expected control objectives. There are different control loops which have different real-time requirements. For instance, the blade pitch angles must be adjusted rapidly.

The WTC has direct economic implications on the LCoE because it has a lasting effect on the AEP and OPEX (cf. Sect. 1.1 and [45]). As a consequence, the most important control objectives are deduced as follows:

- Maximize the energy harvest:

  The controller must maintain the optimum power coefficient by operating the turbine at the optimal tip-speed-ratio λ^* in the PLR. This is especially critical for maximizing AEP.

- Limit the rotor torque:

  The available aerodynamic power in the wind flow exceeds the allowable maximum generator power in the FLR. Thus, the WTC must make sure that the nominal electrical power is not exceeded on average which requires a rapid pitch control actuation (also to limit rotor over speed).

[17] The fatigue reference f_{ref} adjusts the tower damping gain (which puts emphasis on the feedback of the nacelle acceleration $\ddot{x}_{\mathrm{T}}$). $f_{\mathrm{ref}} = 0$ means no damping and $f_{\mathrm{ref}} = 1$ the maximum allowed damping.

- Reduce nacelle oscillations:

  Mechanical vibrations in fore-aft direction correlate with alternating tower bending moments and thus fatigue loads on the tower structure, cf. [34] (p. 492). In order to achieve the desired lifetime of the tower and possibly beyond that, the oscillations must be kept in allowable ranges by active pitch control (mainly in the FLR).

- Minimize yaw misalignments:

  The turbine must face the wind in order to produce energy properly which requires an active yaw controller (less relevant in the FLR). Though, due to the *cosine effect* – meaning $\cos\gamma \approx 1$ for small angles – this is not as time-critical as pitch control and needs to be adjusted only from time to time. Additionally, too much yaw control action would result in unacceptable wear on yaw drive system (increasing the OPEX).

In order to achieve these control objectives, a suitable plant controller must be designed. The WTC used in the present thesis is inspired by the controller architecture proposed by Jonkman et al. [95] which is widely considered as the typical industrial turbine controller, cf. [19] and [185] (p. 47). It consists of two separate controllers for the generator speed n_g which use the generator torque and the collective pitch angle to maximize/limit the generator power. Since this is not sufficient to achieve the third and the fourth objective (see above), the existing baseline controller structure is enhanced by a fatigue controller with variable gain to reduce nacelle oscillations and also by a simplified yaw controller to correct misalignments. Moreover, a power reference input is introduced in order to be able to vary the desired power production.

In summary, there are in total three measurement inputs to the WTC (cf. Fig. 3.9): The generator speed n_g, the nacelle acceleration $\ddot{x}_\mathrm{T}$ and the yaw angle γ_w. In addition to that, there are three reference inputs: The power reference p_ref, the fatigure reference f_ref and the yaw reference γ_ref. These are processed by the control algorithm and produce the desired control actuations. Further details on the controller are omitted due to sake of brevity and due to the availability of extensive literature [34, 22, 140, 17].

3.4 Physical Modeling

The primary goal of this section is the derivation of internal design models for the estimators and filter algorithms, respectively. Secondly, an advanced control-oriented wind turbine model is developed for simulation purposes. The wind turbine models are always divided into a wind field sub-model, an aerodynamic sub-model and an elastodynamic sub-model. Each module may change in level of detail. The required accuracy of the model or application defines its degrees of freedom and the model granularity.

3.4.1 Atmospheric Wind Field Model

The evolution of the wind field in front of the turbine is a complex topic. The full wind field is usually described by a three-dimensional wind vector field

$$\vec{\boldsymbol{v}} = \vec{\boldsymbol{v}}(x,y,z,t) = \begin{bmatrix} u(x,y,z,t) \\ v(x,y,z,t) \\ w(x,y,z,t) \end{bmatrix} = \begin{bmatrix} u(\vec{\boldsymbol{x}},t) \\ v(\vec{\boldsymbol{x}},t) \\ w(\vec{\boldsymbol{x}},t) \end{bmatrix} \tag{3.23}$$

to capture its full complexity. The velocity components of this vector in x, y and z-direction are $u(\vec{\boldsymbol{x}},t)$, $v(\vec{\boldsymbol{x}},t)$ and $w(\vec{\boldsymbol{x}},t)$, respectively. The wind field is therefore an arbitrary vector valued function dependent in time and also three-dimensional space (Fig. 3.11). Thus, the control systems engineer must undertake admissible measures to simplify the wind model and reduce the complexity to a few deterministic parameters. The first step is to consider $x=-\infty$ only, then the three components can be described in the inertial reference frame $\mathcal{I}$ by a so-called hub-height wind field (hh-field) as follows:

$$u_{\mathcal{I}}(y_{\mathcal{I}},z_{\mathcal{I}},t) = \left(\left(\frac{z_{\mathcal{I}}}{H}\right)^{\kappa(t)} + \frac{y_{\mathcal{I}}}{2R}\delta_{\mathrm{h}}(t) + \frac{z_{\mathcal{I}}-H}{2R}\delta_{\mathrm{v}}(t)\right) v_{\mathrm{w}}(t)\cos\alpha_{\mathrm{h}}(t)\cos\alpha_{\mathrm{v}}(t) \tag{3.24a}$$

$$v_{\mathcal{I}}(y_{\mathcal{I}},z_{\mathcal{I}},t) = \left(\left(\frac{z_{\mathcal{I}}}{H}\right)^{\kappa(t)} + \frac{y_{\mathcal{I}}}{2R}\delta_{\mathrm{h}}(t) + \frac{z_{\mathcal{I}}-H}{2R}\delta_{\mathrm{v}}(t)\right) v_{\mathrm{w}}(t)\sin\alpha_{\mathrm{h}}(t)\cos\alpha_{\mathrm{v}}(t) \tag{3.24b}$$

$$w_{\mathcal{I}}(y_{\mathcal{I}},z_{\mathcal{I}},t) = \left(\left(\frac{z_{\mathcal{I}}}{H}\right)^{\kappa(t)} + \frac{y_{\mathcal{I}}}{2R}\delta_{\mathrm{h}}(t) + \frac{z_{\mathcal{I}}-H}{2R}\delta_{\mathrm{v}}(t)\right) v_{\mathrm{w}}(t)\sin\alpha_{\mathrm{v}}(t)\,. \tag{3.24c}$$

This model has been derived and unified based on wind field models proposed in [138, 185]. The time-dependent variables/inputs and constant parameters in Eq. (3.24) are identified

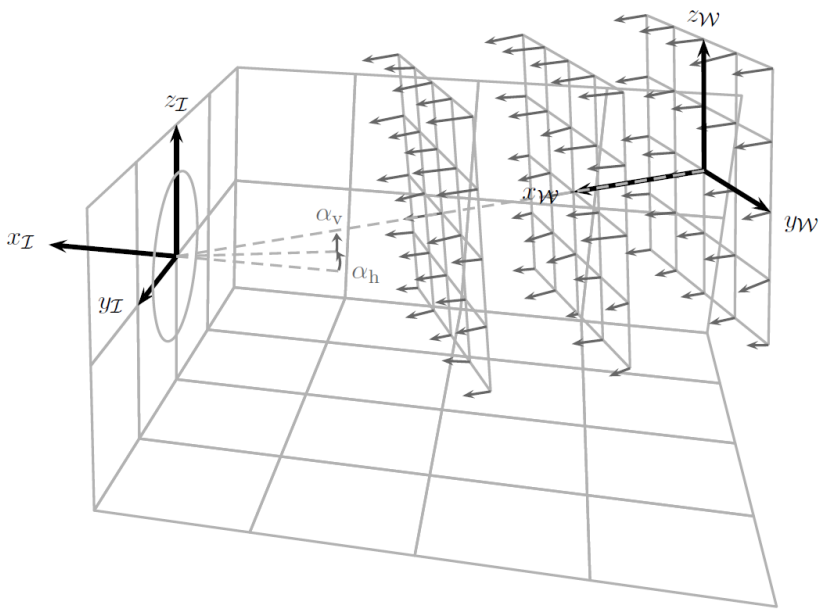

Figure 3.11: Representation of the complex wind field approaching the wind turbine rotor (illustration by David Schlipf, cf. [185] on p. 35)

according to Table 3.2. If the vertical inflow is negligible (and therefore $\alpha_\mathrm{v} = 0$), the model (3.24) reduces to

$$u_\mathcal{I}(y_\mathcal{I}, z_\mathcal{I}, t) = \left(\left(\frac{z_\mathcal{I}}{H}\right)^{\kappa(t)} + \frac{y_\mathcal{I}}{2R}\delta_\mathrm{h}(t) + \frac{z_\mathcal{I}-H}{2R}\delta_\mathrm{v}(t)\right) v_\mathrm{w}(t)\cos\alpha_\mathrm{h}(t) \tag{3.25a}$$

$$v_\mathcal{I}(y_\mathcal{I}, z_\mathcal{I}, t) = \left(\left(\frac{z_\mathcal{I}}{H}\right)^{\kappa(t)} + \frac{y_\mathcal{I}}{2R}\delta_\mathrm{h}(t) + \frac{z_\mathcal{I}-H}{2R}\delta_\mathrm{v}(t)\right) v_\mathrm{w}(t)\sin\alpha_\mathrm{h}(t) \tag{3.25b}$$

and $w_\mathcal{I}(y_\mathcal{I}, z_\mathcal{I}, t) = 0$. Furthermore assuming $\alpha_\mathrm{h} = 0$ results in

$$u_\mathcal{I}(y_\mathcal{I}, z_\mathcal{I}, t) = \left(\left(\frac{z_\mathcal{I}}{H}\right)^{\kappa(t)} + \frac{y_\mathcal{I}}{2R}\delta_\mathrm{h}(t) + \frac{z_\mathcal{I}-H}{2R}\delta_\mathrm{v}(t)\right) v_\mathrm{w}(t) \tag{3.26}$$

and $v_\mathcal{I}(y_\mathcal{I}, z_\mathcal{I}, t) = w_\mathcal{I}(y_\mathcal{I}, z_\mathcal{I}, t) = 0$. Hence, the wind turbine yaw is fully aligned with wind direction ($\gamma = \delta_\mathrm{w}$). The reduced model (3.26) is the simplest spatial wind field model that takes horizontal and vertical shear (linear and exponential) effects into account.

However, it is still not directly applicable as control-oriented model since it is defined for arbitrary coordinates $\{y_\mathcal{I}, z_\mathcal{I}\}$. To overcome this issue, a blade-effective radius $r_\mathrm{B} = r_\mathrm{B}(v_\mathrm{w})$ is proposed where all aerodynamic inflow is assumed to interact with the rotor. Therewith the infinite $\mathbb{R}^2$ space is condensed to a finite number of samples which yields

$$y_{\mathcal{I},1} = -r_\mathrm{B}(v_\mathrm{w})\sin\psi_1(t)\ , \qquad z_{\mathcal{I},1} = H + r_\mathrm{B}(v_\mathrm{w})\cos\psi_1(t) \tag{3.27a}$$

$$y_{\mathcal{I},2} = -r_\mathrm{B}(v_\mathrm{w})\sin\psi_2(t)\ , \qquad z_{\mathcal{I},2} = H + r_\mathrm{B}(v_\mathrm{w})\cos\psi_2(t) \tag{3.27b}$$

$$y_{\mathcal{I},3} = -r_\mathrm{B}(v_\mathrm{w})\sin\psi_3(t)\ , \qquad z_{\mathcal{I},3} = H + r_\mathrm{B}(v_\mathrm{w})\cos\psi_3(t) \tag{3.27c}$$

for a standard three-bladed wind turbine rotor ($n_\mathrm{bl} = 3$). Fig. 3.12 illustrates the situation. Therein, the individual azimuth angle of each blade ψ_b is the defined according to

$$\psi_b = \varphi + 2\pi/3\,(b-1) \quad \text{with} \quad b = \{1, 2, 3\} \tag{3.28}$$

where φ is denoted as the rotor azimuth angle or simply azimuth. The sample points (3.27)

Table 3.2: Input variables and parameters of the atmospheric wind field model (3.24)

Variable	Description
$v_\mathrm{w}(t)$	ambient (absolute) wind speed at hub height H
$\alpha_\mathrm{h}(t)$	horizontal inflow angle (which is the difference between horizontal wind direction δ_w and rotor yaw angle γ)
$\alpha_\mathrm{v}(t)$	vertical inflow angle
$\kappa(t)$	vertical shear exponent (from power law), $\kappa = \{0 \cdots 0.3\}$
$\delta_\mathrm{h}(t)$	linear horizontal shear coefficient
$\delta_\mathrm{v}(t)$	linear vertical shear coefficient
H	hub height (predefined by design)
R	blade tip radius (rotor diameter $D = 2R$, predefined by design)

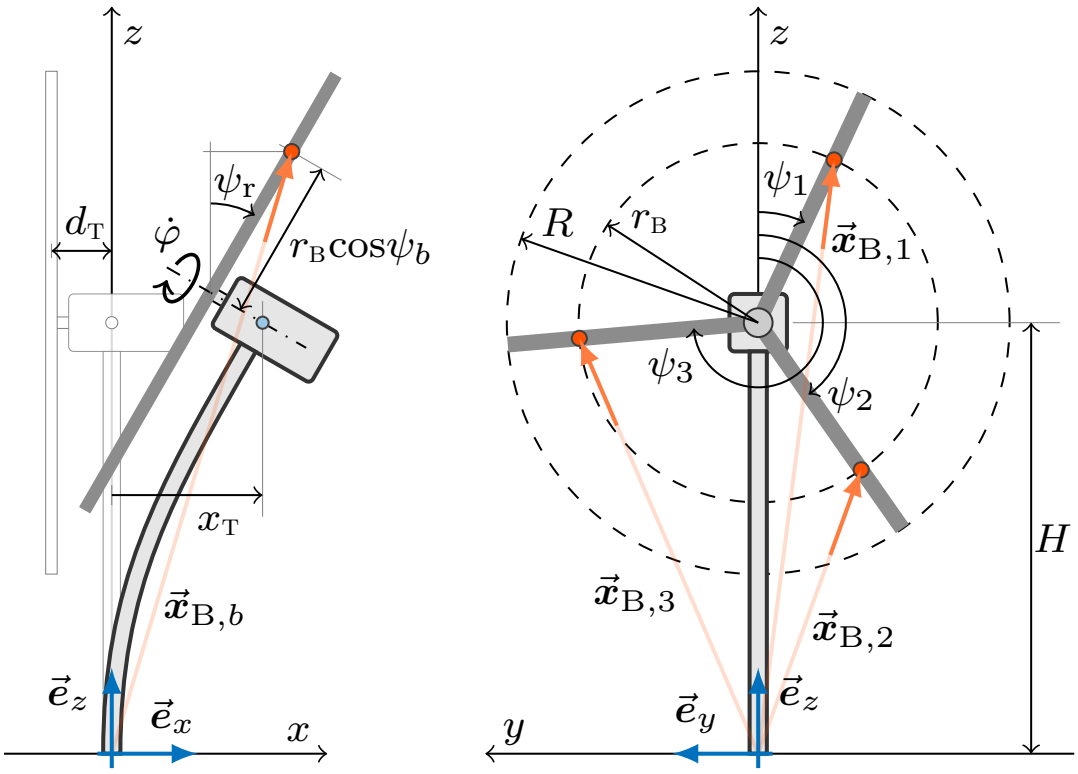

Figure 3.12: Front and side view sketch of the wind turbine including the nacelle and rotor deflection in order to illustrate the effective points on the rotor blades (orange) that are considered for the interaction with the blade-effective wind speed (clock-wise blade rotation)

are inserted in Eq. (3.26) to obtain the vector of blade-effective wind speeds $\boldsymbol{u}_\mathcal{I}(t)$ specified as follows:

$$\boldsymbol{u}_\mathcal{I}(t) = \begin{bmatrix} u_\mathcal{I}(y_{\mathcal{I},1}, z_{\mathcal{I},1}, t) \\ u_\mathcal{I}(y_{\mathcal{I},2}, z_{\mathcal{I},2}, t) \\ u_\mathcal{I}(y_{\mathcal{I},3}, z_{\mathcal{I},3}, t) \end{bmatrix} \in \mathbb{R}^3 \quad . \tag{3.29}$$

For the following derivations the nonlinear exponential law for vertical shear is exempted by choosing $\kappa(t)=0$. Moreover, the explicit time-dependency is dropped for sake of conciseness. Then the atmospheric wind speed model denotes

$$\boldsymbol{u}_\mathcal{I}(t) = \begin{bmatrix} v_\mathrm{w} + \frac{r_\mathrm{B}}{2R}\left(\delta_\mathrm{v} v_\mathrm{w} \cos\psi_1 - \delta_\mathrm{h} v_\mathrm{w} \sin\psi_1\right) \\ v_\mathrm{w} + \frac{r_\mathrm{B}}{2R}\left(\delta_\mathrm{v} v_\mathrm{w} \cos\psi_2 - \delta_\mathrm{h} v_\mathrm{w} \sin\psi_2\right) \\ v_\mathrm{w} + \frac{r_\mathrm{B}}{2R}\left(\delta_\mathrm{v} v_\mathrm{w} \cos\psi_3 - \delta_\mathrm{h} v_\mathrm{w} \sin\psi_3\right) \end{bmatrix} = \begin{bmatrix} 1 & \cos\psi_1 & -\sin\psi_1 \\ 1 & \cos\psi_2 & -\sin\psi_2 \\ 1 & \cos\psi_3 & -\sin\psi_3 \end{bmatrix} \begin{bmatrix} v_\mathrm{w} \\ \frac{r_\mathrm{B}}{2R}\delta_\mathrm{v} v_\mathrm{w} \\ \frac{r_\mathrm{B}}{2R}\delta_\mathrm{h} v_\mathrm{w} \end{bmatrix} \tag{3.30}$$

Interestingly, the above Eq. (3.30) resembles the socalled Park-Transformation[18] if the zero-component is defined as $v_0 = v_\mathrm{w}$, the direct-component as $v_\mathrm{d} = \frac{r_\mathrm{B}}{2R}\delta_\mathrm{v} v_\mathrm{w}$ and the quadrature-component as $v_\mathrm{q} = \frac{r_\mathrm{B}}{2R}\delta_\mathrm{h} v_\mathrm{w}$. Therefore, the individual blade wind speeds are assumed to be

[18] which was invented by Robert H. Park in 1929 and is commonly known from theory of synchronous machines [154]. It is also often denoted as Coleman-Transformation in wind energy science since Coleman & Feingold took up Park's theory in 1943/1958 and applied it successfully to helicopters [42].

generated from a non-rotating frame by Park's Transformation where the shears are defined as

$$\delta_{\mathrm{v}} = \frac{2R}{r_{\mathrm{B}}}\frac{v_{\mathrm{d}}}{v_0} \quad \text{and} \quad \delta_{\mathrm{h}} = \frac{2R}{r_{\mathrm{B}}}\frac{v_{\mathrm{q}}}{v_0} \ . \tag{3.31}$$

However, the d/q-components depend not only on their corresponding shear coefficients but also on the wind speed v_{w} itself. To overcome this problem, an alternative definition of the shears is given by

$$\delta'_{\mathrm{v}}(t) = \frac{\delta_{\mathrm{v}}(t)v_{\mathrm{w}}(t)}{2R} = v_{\mathrm{d}}(t)r_{\mathrm{B}}^{-1} \quad \text{and} \quad \delta'_{\mathrm{h}}(t) = \frac{\delta_{\mathrm{h}}(t)v_{\mathrm{w}}(t)}{2R} = v_{\mathrm{q}}(t)r_{\mathrm{B}}^{-1} \ . \tag{3.32}$$

This approach has also proved to be reasonable for wind field reconstruction using LiDAR [185] since v_{d} and v_{q} then depend linearly on a single variable and not on a product of two independent variables. Both models still agree in principle and are equivalent. However, if δ_{v} is assumed constant, the ratio of $v_{\mathrm{d}}/v_{\mathrm{w}}$ is constant and if δ'_{v} is assumed to be constant, v_{d} is actually independent of the wind speed v_{w}.

3.4.2 Relative Wind Speed Model

So far no wind turbine influences and only natural atmospheric effects have been incorporated in the wind field model $\boldsymbol{u}_{\mathcal{I}}(t)$, see Eq. (3.30). As a matter of fact, the turbine dynamics modify the wind field experienced by the turbine. An illustrative example is the platform pitch motion of floating offshore wind turbines (FOWT) [97, 178]. Due to this influence the rotor sees a modified relative wind speed in front of the turbine. At worst this effect causes instability of the platform dynamics (negative damping) and hence increased loads or possibly destruction even with in non-turbulent wind [114].

As a result, the natural wind field model (3.29) needs to be substituted by a more accurate model. In this thesis, the model

$$\boldsymbol{v}_{\mathrm{rel}} = \mathcal{A}(\varphi)\boldsymbol{u}_{\mathcal{I}}(t) - \boldsymbol{g}(\boldsymbol{x}(t), \boldsymbol{p}, t) \ . \tag{3.33}$$

is proposed which will be further denoted as *relative wind speed model*. The diagonal scaling matrix $\mathcal{A}(\varphi)$ is introduced to take the tower influence on the rotating blades into account. The function $\boldsymbol{g}(\cdot)$ is needed to include the effects of tower and blade dynamic oscillations. The negative sign in Eq. (3.33) is chosen indicatively to account for the defined reference coordinate system. $\mathcal{A}(\varphi)$ and $\boldsymbol{g}(\cdot)$ are derived hereafter.

Tower Dam Static Influence model A generic tower influence model has been proposed originally by Bak et al. [8] which is based on the potential flow solution around a cylinder. In general, the wind turbine's tower blocks the air stream which results in a velocity deficit that is experienced by the passing blades [34] (p. 219 ff). The tower influence has a strong

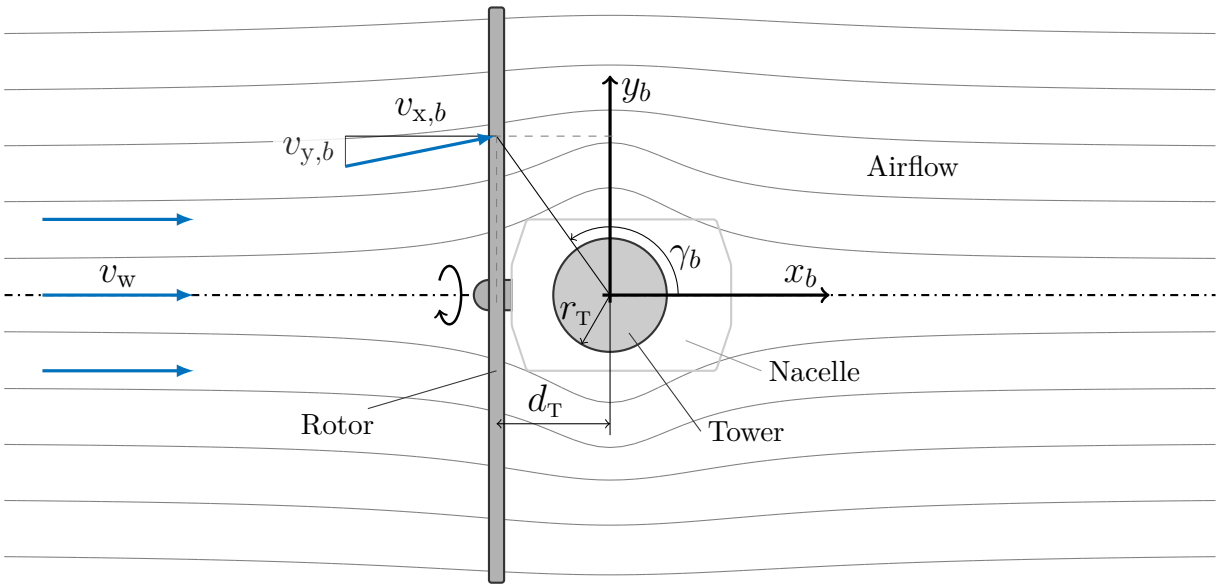

Figure 3.13: Airflow in the vicinity of the tower for rotor blade b (top view) where the nacelle is only implied by the grey line for the sake of better clarity (but is not part of the model)

effect on the 3p turbine loads[19] and the power quality [175] (p. 28/139). The upstream and downstream influences are called tower dam and shadow effect, respectively. For modeling of up-wind turbines only the tower dam effect is considered as relevant (cf. Fig. 3.13).

According to [175, 17], the simplified tower influence model is stated as

$$\frac{v_{x,b}}{v_w} = \begin{cases} 1 - \dfrac{r_T^2}{x_b^2+y_b^2}\cos 2\gamma_b & \text{for} \quad \dfrac{\pi}{2} \le \psi_b < \dfrac{3\pi}{2} \\ 1, & \text{for} \quad -\dfrac{\pi}{2} \le \psi_b < \dfrac{\pi}{2} \end{cases} \tag{3.34a}$$

$$\frac{v_{y,b}}{v_w} = \begin{cases} \dfrac{r_T^2}{x_b^2+y_b^2}\sin 2\gamma_b & \text{for} \quad \dfrac{\pi}{2} \le \psi_b < \dfrac{3\pi}{2} \\ 0, & \text{for} \quad -\dfrac{\pi}{2} \le \psi_b < \dfrac{\pi}{2} \end{cases} \tag{3.34b}$$

wherein $v_{x,b}$ and $v_{y,b}$ are the wind speeds perpendicular and parallel to the rotor plane. Only $v_{x,b}$ is important for the power production (cf. Fig. 3.14) and therefore $v_{y,b}$ is not further considered. The parameter r_T is the cylinder's radius (the tower radius) and the angle γ_b is derived from simple trigonometry as

$$\cos\gamma_b = \frac{y_b}{\sqrt{x_b^2+y_b^2}} \quad \Rightarrow \quad \gamma_b = \arccos\frac{y_b}{\sqrt{x_b^2+y_b^2}}\,, \tag{3.35}$$

[19] The symbol p refers to the multiples of the rotor speed n_r. Thus, 1p means once per revolution and 3p means three times per revolution. A rough estimate for the rotor speed of state-of-the-art turbines is 12 rpm or 0.2 Hz. Hence, the 3p frequency is 0.6 Hz which is often close to the first rotor blade eigenfrequency.

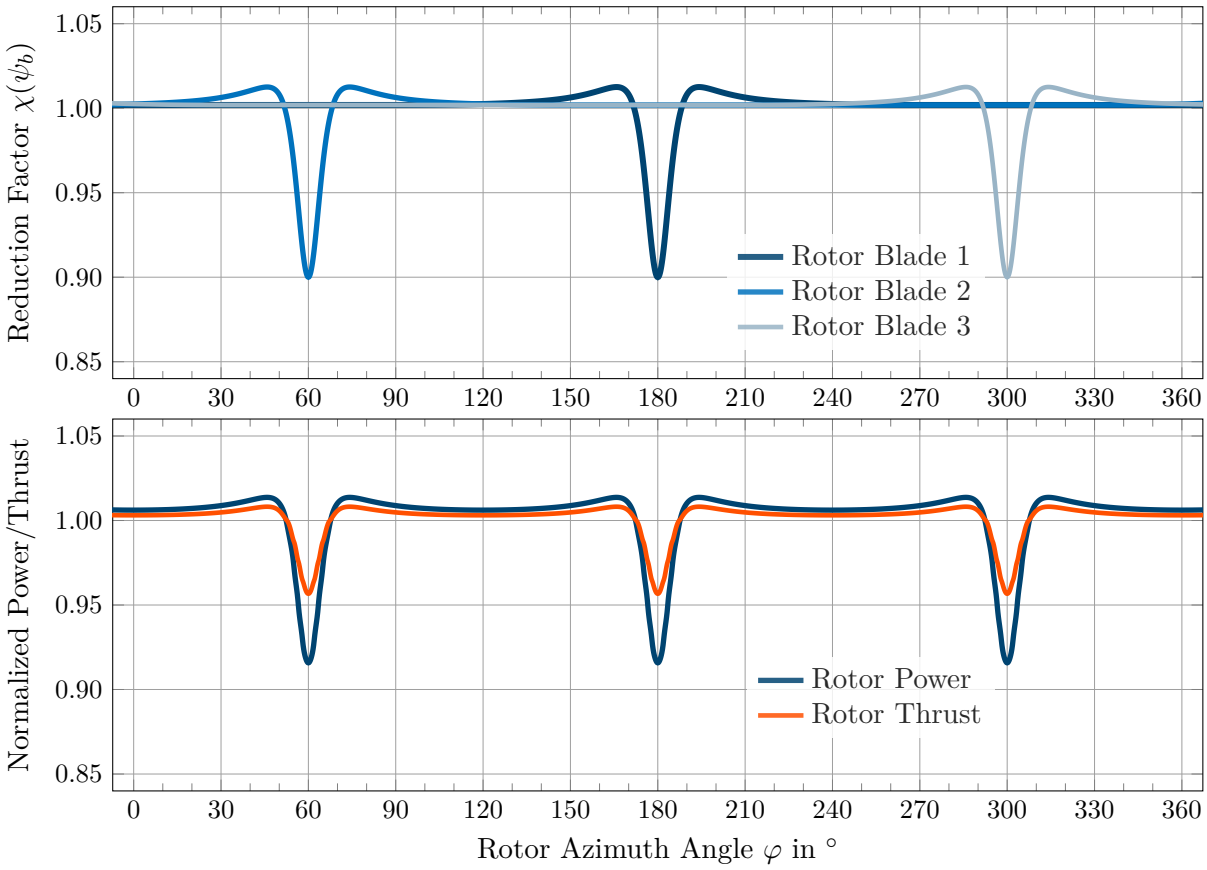

Figure 3.14: Illustration of the tower dam effect for $\mu_T = 0.1$ and $\sigma_T = 50$: the blade velocity deficits at $r_B = \frac{3}{4}R$ (top), the normalized aerodynamic rotor power and thrust (bottom)

see Fig. 3.13. In order to use Eqs. (3.34) in the design model, the angle γ_b must be replaced by the blade azimuth angle ψ_b. For this purpose, the blade coordinates are represented by

$$x_b = -d_T \tag{3.36}$$

$$y_b = -r_B \sin\psi_b \tag{3.37}$$

considering the blade effective radius r_B (Fig. 3.12, p. 62) and the upwind distance d_T as model parameters. Therewith, the sought after relationship $\gamma_b = \gamma_b(\psi_b)$ is obtained. Inserting Eq. (3.35) in Eq. (3.34a) and exploiting the inverse trigonometric formula

$$\cos\Big(2\arccos(x)\Big) = \cos^2\Big(\arccos(x)\Big) - \sin^2\Big(\arccos(x)\Big) = x^2 - (1-x^2) = 2x^2 - 1 \tag{3.38}$$

yields the tower dam reduction factor χ

$$\chi(\psi_b) = \begin{cases} 1 - \mu_T \dfrac{1 - \sigma_T \sin^2\psi_b}{\left(1 + \sigma_T \sin^2\psi_b\right)^2} & \text{for} \quad \dfrac{\pi}{2} \le \psi_b < \dfrac{3\pi}{2} \\[2ex] 1 - \mu_T \dfrac{1 - \sigma_T}{\left(1 + \sigma_T\right)^2}\ , & \text{for} \quad -\dfrac{\pi}{2} \le \psi_b < \dfrac{\pi}{2} \end{cases} \tag{3.39}$$

as a function of only two independent and dimensionless scaling parameters μ_T and σ_T (compared to three parameters before).[20] The first parameter is defined as

$$\mu_\mathrm{T} = \left(\frac{r_\mathrm{T}}{d_\mathrm{T}}\right)^2, \quad \mu_\mathrm{T} \in [0\,,1] \tag{3.40}$$

which determines the maximum velocity deficit $\min\{\chi(\psi_b)\} = 1 - \mu_\mathrm{T}$ for $\psi_b = \pi$. The lower bound in Eq. (3.40) means no tower influence ($\mu_\mathrm{T} = 0$), while the upper bound means wind speed completely cancelled instantaneously ($\mu_\mathrm{T} = 1$). Typical values of μ_T are supposed to lie in between 0.02 and 0.1, therefore reducing the wind speed by up to 10 % at the lowest blade position. The second parameter

$$\sigma_\mathrm{T} = \left(\frac{r_\mathrm{B}}{d_\mathrm{T}}\right)^2, \quad \text{typ. } \sigma_\mathrm{T} \gg 1 \tag{3.41}$$

determines mainly the width of the affected angular range. This is roughly defined by the zeros of the function $1 - \sigma_\mathrm{T} \sin^2\psi_b \stackrel{!}{=} 0$. Since the effective radius r_B is bigger than the distance from yaw center to hub, $r_\mathrm{B}^2 \gg d_\mathrm{T}^2$, typical values for σ_T lie between 10 and 50. Fig. 3.14 on p. 65 shows the reduction factors from Eq. (3.39) for each blade and the effect on the aerodynamic power (both as a function of the rotor azimuth angle).

Finally, the scaling matrix $\mathcal{A}(\varphi)$ for the atmospheric wind model $\boldsymbol{u}_\mathcal{I}$, see Eq. (3.33), is obtained as

$$\mathcal{A}(\varphi) = \begin{bmatrix} \chi(\psi_1) & 0 & 0 \\ 0 & \chi(\psi_2) & 0 \\ 0 & 0 & \chi(\psi_3) \end{bmatrix} \tag{3.42}$$

where ψ_1, ψ_2 and ψ_3 are defined by Eq. (3.28) and $\chi(\psi_b)$ by Eq. (3.39).

Tower and Blade Dynamic Influence model Depending on the level of detail, the vector function $\boldsymbol{g}(\cdot)$ depends on the vector of dynamic states $\boldsymbol{x}(t)$ and also on known model parameters $\boldsymbol{p}$ (in special cases explicitly on time t). Consider first the illustrations in Fig. 3.15 and also Fig. 3.12 (p. 62). As observed therein, the deflections of the tower and the rotor affect directly the position vector $\vec{\boldsymbol{x}}_{\mathrm{B},b}$ of an arbitrary blade b and therefore also its velocity vector. For the relative wind speed computation only the x-component $x_{\mathrm{B},b}$ is considered which is obtained from the inner product with $\vec{\boldsymbol{e}}_x$ according to

$$\vec{\boldsymbol{x}}_{\mathrm{B},b} \cdot \vec{\boldsymbol{e}}_x = x_{\mathrm{B},b} = x_\mathrm{T} + \sin\psi_\mathrm{r}\, r_\mathrm{B} \cos\psi_b - \cos\psi_\mathrm{r}\, d_\mathrm{T} \tag{3.43}$$

wherein ψ_r is the rotor tilt angle, r_B is the blade-effective radius and d_T is the distance from the tower center to the hub. The deflection of the nacelle x_T is considered at hub height H. In order to obtain $\boldsymbol{g}(\cdot)$, the time derivative of Eq. (3.43) has to be evaluated which yields

$$\dot{x}_{\mathrm{B},b} = \dot{x}_\mathrm{T} + \dot{\psi}_\mathrm{r} \cos\psi_\mathrm{r}\, r_\mathrm{B} \cos\psi_b - \sin\psi_\mathrm{r}\, r_\mathrm{B} \dot{\psi}_b \sin\psi_b + \dot{\psi}_\mathrm{r} \sin\psi_\mathrm{r}\, d_\mathrm{T} \tag{3.44}$$

[20] Further note that the Eq. (3.39) is defined slightly different compared to initial Eq. (3.34a) in order to avoid the discontinuity at $\psi_b = \pm\,\pi/2$.

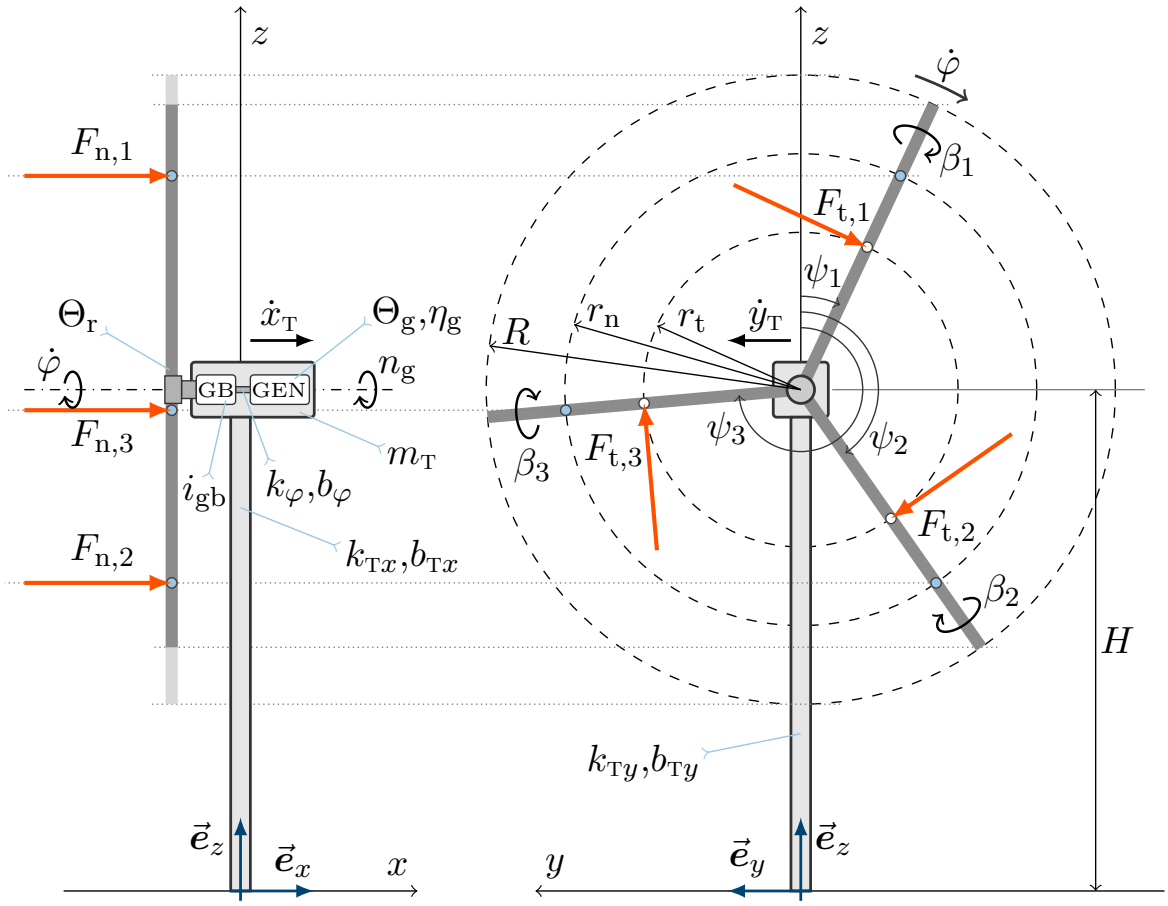

Figure 3.15: Illustration of the abstracted wind turbine model (the gear-box (GB) couples the high-speed-side of the generator (GEN) with the low-speed-side of the rotor via a flexible shaft)

From the theory of bending beams it is known that the following relations

$$\sin\psi_r \approx \psi_r = \frac{3}{2H}x_T = \zeta x_T \ , \quad \dot{\psi}_r = \frac{3}{2H}\dot{x}_T = \zeta \dot{x}_T \ \text{ and } \ \cos\psi_r \approx 1 \tag{3.45}$$

hold when small angles ψ_r are assumed.[21] In Eq. (3.45), ζ is denoted as beam coupling coefficient where $\zeta \ll 1$ for large hub heights H. That means, Eq. (3.44) simplifies to

$$\dot{x}_{B,b} = \dot{x}_T + \zeta \dot{x}_T\, r_B \cos\psi_b - \zeta x_T\, r_B \dot{\psi}_b \sin\psi_b + \zeta^2 \dot{x}_T x_T\, d_T \tag{3.46}$$

where, in addition, the last term is two magnitude orders smaller than the others and thus negligible. For rigid blades, the derivatives of the blade azimuth angles $\dot{\psi}_b$ are then equal to the rotor angular speed $\dot{\varphi} = \Omega$. Hence, for the bottom-fixed horizontal axis wind turbine

$$\boldsymbol{g}(\boldsymbol{x}(t),\boldsymbol{p}) = \begin{bmatrix} \dot{x}_{B,1}(\boldsymbol{x}(t),\boldsymbol{p}) \\ \dot{x}_{B,2}(\boldsymbol{x}(t),\boldsymbol{p}) \\ \dot{x}_{B,3}(\boldsymbol{x}(t),\boldsymbol{p}) \end{bmatrix} = \begin{bmatrix} \big(1+\zeta r_B \cos\psi_1\big)\dot{x}_T - \zeta x_T r_B \dot{\varphi} \sin\psi_1 \\ \big(1+\zeta r_B \cos\psi_2\big)\dot{x}_T - \zeta x_T r_B \dot{\varphi} \sin\psi_2 \\ \big(1+\zeta r_B \cos\psi_3\big)\dot{x}_T - \zeta x_T r_B \dot{\varphi} \sin\psi_3 \end{bmatrix} \tag{3.47}$$

[21] Practically, $\psi_r < 0.07$ (approx. 4°) holds which makes the assumptions in Eq. (3.45) very accurate.

is finally obtained as a simplified tower and blade influence model. The model (3.47) is valid under the assumption of rigid blades and thus approximately negligible blade velocities in flap and edge direction at the radius $r = r_{\mathrm{B}}$.

If the above assumptions are grossly violated, Eq. (3.47) must be enhanced by an elastic rotor blade model which is recommended for advanced control problems.[22] Though, for practical considerations it is mostly sufficiently accurate.

3.4.3 Simplified Aerodynamic Model

The aerodynamic model describes the mathematical relations between the relative wind speed model and the aerodynamic forces and moments acting on the wind turbine rotor. Fig. 3.15 shows a sketch of the mechanical forces.

For the pursued control-oriented modeling approach it is sufficient to employ only quasi-static characteristic which are denoted as aerodynamic coefficients. These are the power coefficient $C_{\mathrm{P}}(\lambda, \beta)$, the torque coefficient $C_{\mathrm{M}}(\lambda, \beta)$ and the rotor thrust coefficient $C_{\mathrm{T}}(\lambda, \beta)$ as shown previously in Fig. 3.7. The dotted line indicates the nominal operating curve of the turbine. The coefficient can either be computed from blade properties like aerodynamic profiles and dimensions [70, 172] or directly from simulation using open source codes like FAST or Aerodyn [96, 138]. Refer to Sect. 3.2.2 for more details.

The aerodynamic coefficients depend mainly on the rotor tip-speed-ratio λ (TSR) and the blade pitch angle β. The influence of the Reynolds number is wittingly neglected (since assumed to be small and to reduce complexity). Moreover, the power and torque coefficient are related by the dimensionless aerodynamic power equation

$$\frac{P_{\mathrm{a}}}{\frac{\varrho}{2} A v_{\mathrm{rel}}^3} = C_{\mathrm{P}}(\lambda, \beta) = \lambda\, C_{\mathrm{M}}(\lambda, \beta) = \frac{\Omega}{v_{\mathrm{rel}}/R} \frac{M_{\mathrm{a}}}{\frac{\varrho}{2} A R v_{\mathrm{rel}}^2} \tag{3.48}$$

where $A = \pi R^2$ denotes the rotor disc swept area and ϱ the air mass density. Eq. (3.48) also constitutes a definition for the aerodynamic rotor power P_{a} and the corresponding aerodynamic torque M_{a}. From the above Eq. (3.48) it becomes apparent why λ is also depictable as *dimensionless rotor speed*.

The power coefficient C_{P} is also directly connected to the thrust coefficient C_{T} via

$$C_{\mathrm{P}}(\lambda, \beta) = 4\, a(\lambda, \beta)\Big(1 - a(\lambda, \beta)\Big)^2 \tag{3.49a}$$

$$C_{\mathrm{T}}(\lambda, \beta) = 4\, a(\lambda, \beta)\Big(1 - a(\lambda, \beta)\Big) \tag{3.49b}$$

since both depend on the axial induction factor $a = a(\lambda, \beta)$. Therefore thrust and power are always linked to each other and in fact power can never be extracted without generating

[22] Such advanced control-oriented rotor blade models have been developed including flap and edge dynamics though are excluded from this thesis for reasons of confidentiality. Illustrative simulation results for state estimation including blade dynamics and blade loads are presented in Sect. 5.5.2 (p. 126).

rotor thrust. Rearranging Eqs. (3.49) provides the relation

$$\frac{C_\mathrm{P}(\lambda,\beta)}{C_\mathrm{T}(\lambda,\beta)} = 1 - a(\lambda,\beta) \tag{3.50}$$

which is helpful to obtain the axial induction from given aerodynamic coefficients. Eq. (3.50) also provides, besides Eq. (3.48), another dimensionless power equation

$$\frac{P_\mathrm{a}}{\frac{\varrho}{2}Av_\mathrm{rel}^3} = C_\mathrm{P}(\lambda,\beta) = \Big(1 - a(\lambda,\beta)\Big)C_\mathrm{T}(\lambda,\beta) = \frac{(1-a)v_\mathrm{rel}\,F_\mathrm{T}}{\frac{\varrho}{2}Av_\mathrm{rel}^3} \tag{3.51}$$

where the aerodynamic power is now derived as product of thrust force F_T and streaming velocity in the rotor plane $(1-a)\,v_\mathrm{rel}$.

The previous explanations have just aimed at a rotor disc interpretation of the wind turbine. However, a more accurate approach is to interpret the rotor as an interaction of three individual blades. Thus, each blade is conceivable as dynamic system which is excited by a tangential force $F_{\mathrm{t},b}$ and normal force $F_{\mathrm{n},b}$ (cf. Fig. 3.15). These forces attack the blade at fictitious tangential and normal equivalent radii. Both, $r_\mathrm{t} = r_\mathrm{t}(v_\mathrm{w})$ and $r_\mathrm{n} = r_\mathrm{n}(v_\mathrm{w})$ are supposed to depend only on the effective wind speed v_w. The following definitions are proposed for individual blade forces:

$$F_{\mathrm{t},b}(v_b,\lambda_b,\beta_b) = \frac{\varrho}{2}\frac{\pi R^3}{3\,r_\mathrm{t}}C_\mathrm{M}(\lambda_b,\beta_b)v_b^2 \tag{3.52a}$$

$$F_{\mathrm{n},b}(v_b,\lambda_b,\beta_b) = \frac{\varrho}{2}\frac{\pi R^2}{3}C_\mathrm{T}(\lambda_b,\beta_b)v_b^2\ . \tag{3.52b}$$

Therein, the individual blade TSR λ_b is specified by

$$\lambda_b = \frac{\dot{\psi}_b R}{v_b} = \frac{\dot{\varphi} R}{v_b} \tag{3.53}$$

and the individual blade pitch angle β_b is given by

$$\beta_b = \beta_\mathrm{c} = \beta \tag{3.54}$$

in case of collective pitch control (CPC) and accordingly

$$\beta_b = \beta_\mathrm{c} + \beta_\mathrm{d}\cos\psi_b + \beta_\mathrm{q}\sin\psi_b \tag{3.55}$$

in case of individual pitch control (IPC), cf. [21]. Both Eqs. (3.52) constitute the foundation to derive the relevant aerodynamic forces and moments for the wind turbine models. Accordingly, the aerodynamic torque M_a and the rotor power P_a are obtained by

$$M_\mathrm{a}(v_b,\lambda_b,\beta_b) = \quad r_\mathrm{t}\sum_{b=1}^{3}F_{\mathrm{t},b}(v_b,\lambda_b,\beta_b) = \frac{\varrho}{2}\frac{\pi R^3}{3}\sum_{b=1}^{3}C_\mathrm{M}(\lambda_b,\beta_b)v_b^2 \tag{3.56a}$$

$$P_\mathrm{a}(v_b,\lambda_b,\beta_b) = \dot{\varphi}\,r_\mathrm{t}\sum_{b=1}^{3}F_{\mathrm{t},b}(v_b,\lambda_b,\beta_b) = \frac{\varrho}{2}\frac{\pi R^2}{3}\sum_{b=1}^{3}C_\mathrm{P}(\lambda_b,\beta_b)v_b^3 \tag{3.56b}$$

which reduce to Eq. (3.48) in case of CPC and collective wind speed assumption $v_b = v_{\mathrm{rel}}$. Moreover, the thrust force F_{T} (in x-coordinate direction) and the sidewise force F_{S} (in y-coordinate direction) denote

$$F_{\mathrm{T}}(v_b, \lambda_b, \beta_b) = \sum_{b=1}^{3} F_{\mathrm{n},b}(v_b, \lambda_b, \beta_b) \qquad = \frac{\varrho}{2}\frac{\pi R^2}{3}\sum_{b=1}^{3} C_{\mathrm{T}}(\lambda_b, \beta_b) v_b^2 \tag{3.57a}$$

$$F_{\mathrm{S}}(v_b, \lambda_b, \beta_b) = \sum_{b=1}^{3} F_{\mathrm{t},b}(v_b, \lambda_b, \beta_b)\cos\psi_b = \frac{\varrho}{2}\frac{\pi R^3}{3\, r_{\mathrm{t}}}\sum_{b=1}^{3} C_{\mathrm{M}}(\lambda_b, \beta_b) v_b^2 \cos\psi_b \ . \tag{3.57b}$$

which is also understood from Fig. 3.15 (p. 67). In addition, the aerodynamic blade root moments in blade-flap and blade-edge direction are attained by

$$M^{\mathrm{a}}_{\mathrm{By},b} = r_{\mathrm{n}}(v_b) F_{\mathrm{n},b}(v_b, \lambda_b, \beta_b) \tag{3.58a}$$

$$M^{\mathrm{a}}_{\mathrm{Bx},b} = r_{\mathrm{t}}(v_b)\, F_{\mathrm{t},b}(v_b, \lambda_b, \beta_b) \tag{3.58b}$$

which then contribute to the nodding and yawing moments on the nacelle as follows:

$$M_{\mathrm{nod}} = \sum_{b=1}^{3} M_{\mathrm{By},b}\cos\psi_b = r_{\mathrm{n}}(v_{\mathrm{w}})\frac{\varrho}{2}\frac{\pi R^2}{3}\sum_{b=1}^{3} C_{\mathrm{T}}(\lambda_b, \beta_b) v_b^2 \cos\psi_b \tag{3.59a}$$

$$M_{\mathrm{yaw}} = \sum_{b=1}^{3} M_{\mathrm{By},b}\sin\psi_b = r_{\mathrm{n}}(v_{\mathrm{w}})\frac{\varrho}{2}\frac{\pi R^2}{3}\sum_{b=1}^{3} C_{\mathrm{T}}(\lambda_b, \beta_b) v_b^2 \sin\psi_b \ . \tag{3.59b}$$

The rotor aerodynamics have a major influence on the wind turbine control. To reduce the complexity for control design a plain static relationship between inflow conditions and aerodynamic forces is sought. This means that a change in the wind field will result in an immediate change of aerodynamic forces and moments.

In the following, the simple and advanced wind turbine models for the filter design and filter implementation are introduced.

3.4.4 Simple Nonlinear Simulation and Design Models

For simple estimation and control tasks it is advantageous to have basic design models at hand. These are suited as low-order internal models and provide insight into the system dynamics with only two nonlinear differential equations. The first dynamic equation comes from the conservation of angular momentum [174, 170, 171]. Applied to the wind turbine's drive-train, it yields

$$\Theta\ddot{\varphi} + \mathcal{B}\dot{\varphi} = M_{\mathrm{a}}(\boldsymbol{v}_{\mathrm{rel}}, \dot{\varphi}, \beta) - i_{\mathrm{gb}} M_{\mathrm{g}} \tag{3.60}$$

where $\Theta = \Theta_{\mathrm{r}} + \Theta_{\mathrm{g}} i_{\mathrm{gb}}^2$ is the equivalent drive-train inertia, i_{gb} is the gear-box ratio and $\mathcal{B}$ is the linear external damping coefficient. On the right side of Eq. (3.60) the driving torque M_{a} is compared to the generator breaking torque M_{g} (which constitutes the so-called *torque balance*).

The second dynamic equation is obtained from conservation of linear momentum (also denoted as *force balance*) for the rotor/nacelle. It reads

$$m_{\mathrm{T}}\ddot{x}_{\mathrm{T}} + b_{\mathrm{T}x}\dot{x}_{\mathrm{T}} + k_{\mathrm{T}x}x_{\mathrm{T}} = F_{\mathrm{T}}(\boldsymbol{v}_{\mathrm{rel}}, \dot{\varphi}, \beta) \tag{3.61}$$

where m_{T} is the equivalent (first) modal mass of the nacelle/tower/rotor vibration system, $b_{\mathrm{T}x}$ is the linear damping coefficient and $k_{\mathrm{T}x}$ is the mechanical stiffness of the tower. The basic design model considers only the rotor thrust F_{T} as inhomogeneity on the right side of Eq. (3.61). The relative wind speed vector $\boldsymbol{v}_{\mathrm{rel}}$ which appears in both Eqs. (3.60) and (3.61) is derived from Eq. (3.33) and simplifies for the basic models as follows

$$\boldsymbol{v}_{\mathrm{rel}} = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = \begin{bmatrix} \chi(\psi_1)\, v_{\mathrm{w}}(t) - \dot{x}_{\mathrm{T}} \\ \chi(\psi_2)\, v_{\mathrm{w}}(t) - \dot{x}_{\mathrm{T}} \\ \chi(\psi_3)\, v_{\mathrm{w}}(t) - \dot{x}_{\mathrm{T}} \end{bmatrix} \quad \text{or} \quad v_b = \chi(\psi_b)\, v_{\mathrm{w}}(t) - \dot{x}_{\mathrm{T}}\ , \quad b = \{1, 2, 3\}\ . \tag{3.62}$$

Hence, only the tower dam effect $\chi(\psi_b)$ produces periodic asymmetries in the wind field, interacting with the rotor-nacelle system. This effect is cancelled by setting $\mu_{\mathrm{T}} = 0$ and thus $\chi(\psi_b) = 1$ holds. In order to reduce the complexity, a plain static relationship between wind inflow v_{in} and the effective wind speed v_{w} is assumed (for the simple models both are equivalent).

The following basic design models are derived from the above equations:

- The Simple Rotor Model 3.1 (see p. 73)

  This model includes only the rotor-generator dynamics with an inelastic drive-train. No nacelle dynamics are considered. Such a model is useful for rudimentary wind turbine simulation studies and for basic power production control tasks.

- The Simple Rotor-Nacelle Model 3.2 (see p. 73)

  This model includes the simple rotor-nacelle dynamics with an inelastic drive-train. No nacelle side-side dynamics are included. Such a model is useful for enhanced wind turbine simulation (considering the tower base bending moment) and for simple power production control including tower fore-aft vibration control.

- The Simple Augmented Rotor Model 3.3 (see p. 74)

  This model includes only the rotor-generator dynamics with an inelastic drive-train. No nacelle dynamics are considered. Such a model is useful for simple wind speed estimation and rotor torque estimation purposes. The model is augmented by a disturbance model for the effective wind speed.

- The Simple Augmented Rotor-Nacelle Model 3.4 (see p. 74)

  This model includes the simple rotor-nacelle dynamics with an inelastic drive-train. No nacelle side-side dynamics are included. Such a model is useful for rotor/nacelle state estimation and wind speed estimation. Moreover, the rotor torque and tower base

bending moment as well as the first eigenfrequency may be estimated. The model is augmented by a disturbance/parameter model for the effective wind speed and the eigenfrequency.

The general assumptions for all these models are

- collective pitch control (CPC),
- a simple rotor model without additional drive-train degrees-of-freedom,
- an effective wind speed model with tower dam effect,
- no vertical or horizontal wind shear effects considered,
- no side-side nacelle oscillations included, and
- no dynamic inflow models.

The more advanced control-oriented models are introduced hereafter.

3.4.5 Advanced Nonlinear Simulation and Design Models

For more advanced estimation and control tasks it is advantageous to have more detailed design models at hand which include further degrees-of-freedom and more dynamic states.

In the present thesis, such an advanced model must include the axial and lateral nacelle motion as well as the dynamics for generator and torsional drive-train oscillations. With four degrees-of-freedom (dof) this gives an eighth-order nonlinear state-space model where each rotor blade contributes separately to the aerodynamic forces, contrary to Model 3.2 (p. 73). The advanced wind turbine models consist therefore of at least four nonlinear second order differential equations including

$$m_{\mathrm{T}}\ddot{x}_{\mathrm{T}} + b_{\mathrm{T}x}\dot{x}_{\mathrm{T}} + k_{\mathrm{T}x}x_{\mathrm{T}} = F_{\mathrm{T}}(v_b, \lambda_b, \beta_b) + \zeta M_{\mathrm{nod}}(v_b, \lambda_b, \beta_b) \tag{3.63}$$

$$-m_{\mathrm{T}}\ddot{y}_{\mathrm{T}} - b_{\mathrm{T}y}\dot{y}_{\mathrm{T}} - k_{\mathrm{T}y}y_{\mathrm{T}} = F_{\mathrm{S}}(v_b, \lambda_b, \beta_b) + \zeta i_{\mathrm{gb}} M_{\mathrm{g}} \tag{3.64}$$

$$\Theta_{\mathrm{r}}(\ddot{\varphi}_{\mathrm{g}} + \Delta\ddot{\varphi}) + \Theta_{\mathrm{g}} i_{\mathrm{gb}}^2 \ddot{\varphi}_{\mathrm{g}} = M_{\mathrm{a}}(v_b, \lambda_b, \beta_b) - i_{\mathrm{gb}} M_{\mathrm{g}} \tag{3.65}$$

$$\Theta_{\mathrm{r}}(\ddot{\varphi}_{\mathrm{g}} + \Delta\ddot{\varphi}) + b_\varphi \Delta\dot{\varphi} + k_\varphi \Delta\varphi = M_{\mathrm{a}}(v_b, \lambda_b, \beta_b) \tag{3.66}$$

Therein, the aerodynamic thrust force $F_{\mathrm{T}}(\cdot)$ is defined in Eq. (3.57a) and side-side force $F_{\mathrm{S}}(\cdot)$ in Eq. (3.57b). Moreover, the aerodynamic torque $M_{\mathrm{a}}(\cdot)$ is defined in Eq. (3.56a) and nodding moment $M_{\mathrm{nod}}(\cdot)$ in Eq. (3.59a). y_{T} represents the lateral position of the nacelle and $\Delta\varphi$ is the drive-train torsional angle – assuming a simple spring-damper-system coupling the two rotating masses of the rotor Θ_{r} and generator Θ_{g}.

The wind model includes the tower dam effect, the nacelle fore-aft motion as well as the shears (linear vertical and linear horizontal):

$$v_b = \chi(\psi_b) v_{\mathrm{w}} + \frac{r_{\mathrm{B}}}{2R}\Big(\delta'_{\mathrm{v}} \cos\psi_b + \delta'_{\mathrm{h}} \sin\psi_b\Big) - \Big(1 + \zeta r_{\mathrm{B}} \cos\psi_b\Big)\dot{x}_{\mathrm{T}} \tag{3.67}$$

Model 3.1 (The Simple Rotor Model) *This model includes only the rotor-generator dynamics with an inelastic drive-train. No nacelle dynamics are considered. Such a model is useful for rudimentary wind turbine simulation studies and for basic power production control tasks.*

The Simple Rotor Model – Variables, Definitions and Equations

Vector Notations:
$$\boldsymbol{x} = \dot{\varphi}_{\mathrm{g}}\,,\ \boldsymbol{u} = \Big[M_{\mathrm{g}}\ \beta\Big]^{\mathrm{T}},\ \boldsymbol{d} = d_1 = v_{\mathrm{w}}\,,\ \boldsymbol{p} = \varphi\,,$$
$$\boldsymbol{y} = n_{\mathrm{g}}\,,\ \boldsymbol{z} = \Big[M_{\mathrm{a}}\ P_{\mathrm{g}}\Big]^{\mathrm{T}}$$

State Equation:
$$\dot{x}_1 = \frac{\varrho}{2}\frac{\pi R^3}{3\Theta}\sum_{b=1}^{3} C_{\mathrm{M}}(\lambda_b, u_2)\big(\chi(\psi_b)d_1\big)^2 - \frac{i_{\mathrm{gb}}}{\Theta}u_1 - \frac{\mathcal{B}}{\Theta}x_1$$

Output Equation:
$$y_1 = 30\,\pi^{-1} i_{\mathrm{gb}} x_1$$

Performance Eqs.:
$$z_1 = \frac{\varrho}{2}\frac{\pi R^3}{3}\sum_{b=1}^{3} C_{\mathrm{M}}(\lambda_b, u_2)\big(\chi(\psi_b)d_1\big)^2\,,\ z_2 = \eta_{\mathrm{g}} i_{\mathrm{gb}} x_1 u_1$$

Definitions:
$$\lambda_b = \frac{x_1 R}{\chi(\psi_b)d_1}\,,\ \psi_b = p_1 + \frac{2\pi}{3}(b-1)$$
$$\Theta = \Theta_{\mathrm{r}} + \Theta_{\mathrm{g}} i_{\mathrm{gb}}^2\,,\ i_{\mathrm{gb}} = \frac{n_{\mathrm{g}}}{n_{\mathrm{r}}} = \frac{2\pi}{60}\frac{n_{\mathrm{g}}}{\dot{\varphi}}$$

Model 3.2 (The Simple Rotor-Nacelle Model) *This model includes the simple rotor-nacelle dynamics with an inelastic drive-train. No nacelle side-side dynamics are included. Such a model is e.g. useful for enhanced wind turbine simulation (considering the tower base bending moment).*

The Simple Rotor-Nacelle Model – Variables, Definitions and Equations

Vector Notations:
$$\boldsymbol{x} = \Big[\dot{\varphi}\ \dot{x}_{\mathrm{T}}\ x_{\mathrm{T}}\Big]^{\mathrm{T}},\ \boldsymbol{u} = \Big[M_{\mathrm{g}}\ \beta\Big]^{\mathrm{T}},\ \boldsymbol{d} = d_1 = v_{\mathrm{w}}\,,\ \boldsymbol{p} = \varphi\,,$$
$$\boldsymbol{y} = \Big[n_{\mathrm{g}}\ \ddot{x}_{\mathrm{T}}\Big]^{\mathrm{T}},\ \boldsymbol{z} = \Big[M_{\mathrm{a}}\ P_{\mathrm{g}}\ M_{\mathrm{T}y}\Big]^{\mathrm{T}}$$

State Equations:
$$\dot{x}_1 = \frac{\varrho}{2}\frac{\pi R^3}{3\Theta}\sum_{b=1}^{3} C_{\mathrm{M}}(\lambda_b, u_2)\Big(\chi(\psi_b)d_1 - x_2\Big)^2 - \frac{i_{\mathrm{gb}}}{\Theta}u_1 - \frac{\mathcal{B}}{\Theta}x_1$$
$$\dot{x}_2 = \frac{\varrho}{2}\frac{\pi R^2}{3m_{\mathrm{T}}}\sum_{b=1}^{3} C_{\mathrm{T}}(\lambda_b, u_2)\Big(\chi(\psi_b)d_1 - x_2\Big)^2 - 2\zeta_{\mathrm{T}}\omega_0 x_2 - \omega_0^2 x_3$$
$$\dot{x}_3 = x_2$$

Output Equations:
$$y_1 = 30\,\pi^{-1} i_{\mathrm{gb}} x_1$$
$$y_2 = \frac{\varrho}{2}\frac{\pi R^2}{3m_{\mathrm{T}}}\sum_{b=1}^{3} C_{\mathrm{T}}(\lambda_b, u_2)\Big(\chi(\psi_b)d_1 - x_2\Big)^2 - 2\zeta_{\mathrm{T}}\omega_0 x_2 - \omega_0^2 x_3$$

Performance Eqs.:
$$z_1 = \frac{\varrho}{2}\frac{\pi R^3}{3}\sum_{b=1}^{3} C_{\mathrm{M}}(\lambda_b, u_2)\Big(\chi(\psi_b)d_1 - x_2\Big)^2$$
$$z_2 = \eta_{\mathrm{g}} i_{\mathrm{gb}} x_1 u_1\,,\ z_3 = \big(2\zeta_{\mathrm{T}}\omega_0 x_2 + \omega_0^2 x_3\big)m_{\mathrm{T}} H$$

Definitions:
$$\lambda_b = \frac{x_1 R}{\chi(\psi_b)d_1 - x_2}\,,\ \psi_b = p_1 + \frac{2\pi}{3}(b-1)\,,\ \omega_0^2 = \frac{k_{\mathrm{T}x}}{m_{\mathrm{T}}}\,,$$
$$\Theta = \Theta_{\mathrm{r}} + \Theta_{\mathrm{g}} i_{\mathrm{gb}}^2\,,\ i_{\mathrm{gb}} = \frac{n_{\mathrm{g}}}{n_{\mathrm{r}}} = \frac{2\pi}{60}\frac{n_{\mathrm{g}}}{\dot{\varphi}}$$

Model 3.3 (The Simple Augmented Rotor Model) *This model includes the rotor-generator dynamics with an inelastic drive-train. No nacelle dynamics are considered. Such a model is useful for simple wind speed estimation and rotor torque estimation purposes.*

The Simple Augmented Rotor Model – Variables, Definitions and Equations

Vector Notations:
$$\boldsymbol{x} = \begin{bmatrix}\dot{\varphi}_\mathrm{g} \; v_\mathrm{w}\end{bmatrix}^\mathrm{T}, \; \boldsymbol{u} = \begin{bmatrix}M_\mathrm{g} \; \beta\end{bmatrix}^\mathrm{T}, \; \boldsymbol{d} = [\,], \; \boldsymbol{p} = \varphi,$$
$$\boldsymbol{y} = n_\mathrm{g}, \; \boldsymbol{z} = \begin{bmatrix}M_\mathrm{a} \; P_\mathrm{g}\end{bmatrix}^\mathrm{T}$$

State Equation:
$$\dot{x}_1 = \frac{\varrho}{2}\frac{\pi R^3}{3\Theta}\sum_{b=1}^{3} C_\mathrm{M}(\lambda_b, u_2)\big(\chi(\psi_b)x_2\big)^2 - \frac{i_\mathrm{gb}}{\Theta}u_1 - \frac{\mathcal{B}}{\Theta}x_1$$
$$\dot{x}_2 = 0$$

Output Equation:
$$y_1 = 30\,\pi^{-1} i_\mathrm{gb} x_1$$

Performance Eqs.:
$$z_1 = \frac{\varrho}{2}\frac{\pi R^3}{3}\sum_{b=1}^{3} C_\mathrm{M}(\lambda_b, u_2)\big(\chi(\psi_b)x_2\big)^2, \; z_2 = \eta_\mathrm{g} i_\mathrm{gb} x_1 u_1$$

Definitions:
$$\lambda_b = \frac{x_1 R}{\chi(\psi_b)x_2}, \; \psi_b = \varphi + \frac{2\pi}{3}(b-1), \; i_\mathrm{gb} = \frac{n_\mathrm{g}}{n_\mathrm{r}} = \frac{2\pi}{60}\frac{n_\mathrm{g}}{\dot{\varphi}}$$
$$\Theta = \Theta_\mathrm{r} + \Theta_\mathrm{g} i_\mathrm{gb}^2$$

Model 3.4 (The Simple Augmented Rotor-Nacelle Model) *This model includes the simple rotor-nacelle dynamics with an inelastic drive-train. No nacelle side-side dynamics are included. Such a model is e.g. useful for rotor/nacelle state estimation and wind speed estimation.*

The Simple Augmented Rotor-Nacelle Model – Variables, Definitions and Equations

Vector Notations:
$$\boldsymbol{x} = \begin{bmatrix}\dot{\varphi} \; \dot{x}_\mathrm{T} \; x_\mathrm{T} \; v_\mathrm{w} \; \omega_0\end{bmatrix}^\mathrm{T}, \; \boldsymbol{u} = \begin{bmatrix}M_\mathrm{g} \; \beta\end{bmatrix}^\mathrm{T}, \; \boldsymbol{d} = [\,], \; \boldsymbol{p} = \varphi,$$
$$\boldsymbol{y} = \begin{bmatrix}n_\mathrm{g} \; \ddot{x}_\mathrm{T}\end{bmatrix}^\mathrm{T}, \; \boldsymbol{z} = \begin{bmatrix}M_\mathrm{a} \; P_\mathrm{g} \; M_{\mathrm{T}y}\end{bmatrix}^\mathrm{T}$$

State Equations:
$$\dot{x}_1 = \frac{\varrho}{2}\frac{\pi R^3}{3\Theta}\sum_{b=1}^{3} C_\mathrm{M}(\lambda_b, u_2)\Big(\chi(\psi_b)x_4 - x_2\Big)^2 - \frac{i_\mathrm{gb}}{\Theta}u_1 - \frac{\mathcal{B}}{\Theta}x_1$$
$$\dot{x}_2 = \frac{\varrho}{2}\frac{\pi R^2}{3m_\mathrm{T}}\sum_{b=1}^{3} C_\mathrm{T}(\lambda_b, u_2)\Big(\chi(\psi_b)x_4 - x_2\Big)^2 - 2\zeta_{\mathrm{T}x}x_5x_2 - x_5^2 x_3$$
$$\dot{x}_3 = x_2, \; \dot{x}_4 = 0, \; \dot{x}_5 = 0$$

Output Equations:
$$y_1 = 30\,\pi^{-1} i_\mathrm{gb} x_1$$
$$y_2 = \frac{\varrho}{2}\frac{\pi R^2}{3m_\mathrm{T}}\sum_{b=1}^{3} C_\mathrm{T}(\lambda_b, u_2)\Big(\chi(\psi_b)x_4 - x_2\Big)^2 - 2\zeta_{\mathrm{T}x}x_5x_2 - x_5^2 x_3$$

Performance Eqs.:
$$z_1 = \frac{\varrho}{2}\frac{\pi R^3}{3}\sum_{b=1}^{3} C_\mathrm{M}(\lambda_b, u_2)\Big(\chi(\psi_b)x_4 - x_2\Big)^2$$
$$z_2 = \eta_\mathrm{g} i_\mathrm{gb} x_1 u_1, \; z_3 = (2\zeta_{\mathrm{T}x}x_5x_2 + x_5^2 x_3)m_\mathrm{T}H$$

Definitions:
$$\lambda_b = \frac{x_1 R}{\chi(\psi_b)d_1 - x_2}, \; \psi_b = \varphi + \frac{2\pi}{3}(b-1)$$
$$\Theta = \Theta_\mathrm{r} + \Theta_\mathrm{g} i_\mathrm{gb}^2, \; i_\mathrm{gb} = \frac{n_\mathrm{g}}{n_\mathrm{r}} = \frac{2\pi}{60}\frac{n_\mathrm{g}}{\dot{\varphi}}, \; \omega_0^2 = \frac{k_{\mathrm{T}x}}{m_\mathrm{T}}, \; 2\zeta_{\mathrm{T}x}\omega_0 = \frac{b_{\mathrm{T}x}}{m_\mathrm{T}}$$

where the blade azimuth angle is $\psi_b = \varphi_\mathrm{g} + \Delta\varphi + 2\pi/3(b-1)$ and the blade tip-speed-ratio is $\lambda_b = (\dot{\varphi}_\mathrm{g} + \Delta\dot{\varphi}) R\, v_b^{-1}$. Moreover, a simplified nonlinear inflow model for the disturbance input v_w with two dynamic states

$$T_\mathrm{w}(v_\mathrm{m})\, \dot{v}_\mathrm{m} = v_\mathrm{in} - v_\mathrm{m} \tag{3.68}$$

$$T_\mathrm{in}(v_\mathrm{m})\, \dot{v}_\mathrm{in} = v_\mathrm{w} - v_\mathrm{in} \tag{3.69}$$

is included which takes the time lag between the wind and the resulting forces (acting on the individual blades) into account. v_m is the mean wind speed and v_in is the delayed inflow wind speed. Both time constants depend on the average wind speed v_m whereas the second relates to the turbulence length scale L as follows, cf. [108]:

$$T_\mathrm{in}(v_\mathrm{m}) = \frac{2L}{\pi} v_\mathrm{m}^{-1}\,. \tag{3.70}$$

Based on the above equations, the following advanced design models are derived:

- The Advanced Rotor-Nacelle Model 3.5 (see p. 76)

  This model includes the interactions between the drive-train and the nacelle dynamics in axial and lateral direction. Such a model is useful as an advanced simulation model and also for controller testing and design.

- The Advanced Augmented Rotor-Nacelle Model 3.6 (see p. 77)

  This model includes the drive-train and nacelle dynamics in axial and lateral direction like Model 3.5. Though some uncertain parameters and the effective wind have been augmented to the state vector in order to be able to estimate/predict these quantities simultaneously. Moreover, additional performance outputs have been defined to estimate mechanical loads at the same time. Such a model is therefore useful for state, parameter and disturbance estimation using a monolithic estimator.

The general features for these models are

- suitability for individual pitch control (IPC),
- an enhanced rotor/generator model including simplified drive-train dynamics,
- an effective wind speed model with tower dam effect,
- linear vertical and horizontal wind shear effects considered,
- side-side nacelle oscillations also included, and
- a simple dynamic inflow model.

Model 3.5 (The Advanced Rotor-Nacelle Model) *The model considers the simplified rotor-generator interaction coupled with the nacelle fore-aft dynamics. Such a model is useful for the simultaneous estimation of tower dynamics, eigenfrequency estimation and wind field estimation.*

The Advanced Rotor-Nacelle Model – Variables, Definitions and Equations

Vector Notations:

$$\boldsymbol{x} = \left[\dot{x}_\mathrm{T}\ \dot{y}_\mathrm{T}\ \dot{\varphi}_\mathrm{g}\ \Delta\dot{\varphi}\ x_\mathrm{T}\ y_\mathrm{T}\ \varphi_\mathrm{g}\ \Delta\varphi\ v_\mathrm{m}\ v_\mathrm{in}\right]^\mathrm{T},\ \boldsymbol{u} = \left[M_\mathrm{g}\ \beta_1\ \beta_2\ \beta_3\right]^\mathrm{T}$$

$$\boldsymbol{d} = \left[v_\mathrm{w}\ \delta'_\mathrm{v}\ \delta'_\mathrm{h}\right]^\mathrm{T},\ \boldsymbol{y} = \left[\ddot{x}_\mathrm{T}\ \ddot{y}_\mathrm{T}\ n_\mathrm{g}\ \varphi\right]^\mathrm{T},\ \boldsymbol{z} = \left[M_\mathrm{a}\ P_\mathrm{g}\ M_{\mathrm{T}y}\right]^\mathrm{T}$$

State Equations:

$$\dot{x}_1 = \frac{\varrho}{6}\frac{\pi R^2}{m_\mathrm{T}}\sum_{b=1}^{3}\Big(1+\zeta r_\mathrm{n}\cos\psi_b\Big)C_\mathrm{T}(\lambda_b, u_{b+1})v_b^2 - 2\zeta_{\mathrm{T}x}\omega_0 x_1 - \omega_0^2 x_5$$

$$\dot{x}_2 = -\frac{\varrho}{6}\frac{\pi R^2}{m_\mathrm{T} r_\mathrm{t}}\sum_{b=1}^{3}\cos\psi_b\, C_\mathrm{M}(\lambda_b, u_{b+1})v_b^2 - \frac{\zeta i_\mathrm{gb}}{m_\mathrm{T}}u_1 - 2\zeta_{\mathrm{T}y}\omega_0 x_2 - \omega_0^2 x_6$$

$$\dot{x}_3 = \frac{1}{\Theta_\mathrm{g} i_\mathrm{gb}^2}(b_\varphi x_4 + k_\varphi x_8 - i_\mathrm{gb}u_1)$$

$$\dot{x}_4 = \frac{\varrho}{6}\frac{\pi R^3}{\Theta_\mathrm{r}}\sum_{b=1}^{3}C_\mathrm{M}(\lambda_b, u_{b+1})v_b^2 + \frac{u_1}{\Theta_\mathrm{g} i_\mathrm{gb}} - \frac{\Theta_\mathrm{r}+\Theta_\mathrm{g} i_\mathrm{gb}^2}{\Theta_\mathrm{r}\Theta_\mathrm{g} i_\mathrm{gb}^2}\Big(b_\varphi x_4 + k_\varphi x_8\Big)$$

$$\dot{x}_5 = x_1\,,\ \dot{x}_6 = x_2\,,\ \dot{x}_7 = x_3\,,\ \dot{x}_8 = x_4$$

$$\dot{x}_9 = T_\mathrm{w}^{-1}(x_{10} - x_9)$$

$$\dot{x}_{10} = \frac{\pi}{2L}x_9(\chi(\psi_b)d_1 - x_{10})$$

Output Equations:

$$y_1 = \frac{\varrho}{6}\frac{\pi R^2}{m_\mathrm{T}}\sum_{b=1}^{3}\Big(1+\zeta r_\mathrm{n}\cos\psi_b\Big)C_\mathrm{T}(\lambda_b, u_{b+1})v_b^2 - 2\zeta_{\mathrm{T}x}\omega_0 x_1 - \omega_0^2 x_5$$

$$y_2 = \frac{\varrho}{6}\frac{\pi R^2}{m_\mathrm{T} r_\mathrm{t}}\sum_{b=1}^{3}\cos\psi_b\, C_\mathrm{M}(\lambda_b, u_{b+1})v_b^2 + \frac{\zeta i_\mathrm{gb}}{m_\mathrm{T}}u_1 + 2\zeta_{\mathrm{T}y}\omega_0 x_2 + \omega_0^2 x_6$$

$$y_3 = 30\,\pi^{-1} i_\mathrm{gb} x_1$$

$$y_4 = x_7 + x_8$$

Performance Eqs.:

$$z_1 = \frac{\varrho}{2}\frac{\pi R^3}{3}\sum_{b=1}^{3}C_\mathrm{M}(\lambda_b, u_{b+1})v_b^2$$

$$z_2 = \eta_\mathrm{g} i_\mathrm{gb} x_1 u_1\,,\ z_3 = (2\zeta_{\mathrm{T}x}\omega_0 x_1 + \omega_0^2 x_5)m_\mathrm{T} H$$

Definitions:

$$\lambda_b = \frac{(x_3+x_4)R}{v_b}\,,\ \omega_0^2 = \frac{k_{\mathrm{T}x}}{m_\mathrm{T}}\,,\ 2\zeta_{\mathrm{T}x}\omega_0 = \frac{b_{\mathrm{T}x}}{m_\mathrm{T}}\,,\ i_\mathrm{gb} = \frac{n_\mathrm{g}}{n_\mathrm{r}} = \frac{2\pi}{60}\frac{n_\mathrm{g}}{\dot{\varphi}}$$

$$v_b = x_{10} + \frac{r_\mathrm{B}}{2R}\Big(d_2\cos\psi_b + d_3\sin\psi_b\Big) - \Big(1+\zeta r_\mathrm{B}\cos\psi_b\Big)x_1$$

$$\psi_b = x_7 + x_8 + \frac{2\pi}{3}(b-1)$$

Model 3.6 (The Advanced Augmented Rotor-Nacelle Model) *The model considers the simplified rotor-generator interaction coupled with the nacelle fore-aft dynamics. Such a model is useful for the simultaneous estimation of tower dynamics, eigenfrequency and wind estimation.*

The Adv. Augmented Rotor-Nacelle Model – Variables, Definitions and Equations

Vector Notations:

$$\boldsymbol{x} = \begin{bmatrix} \dot{x}_\mathrm{T} & \dot{y}_\mathrm{T} & \dot{\varphi}_\mathrm{g} & \Delta\dot{\varphi} & x_\mathrm{T} & y_\mathrm{T} & \varphi_\mathrm{g} & \Delta\varphi & v_\mathrm{m} & v_\mathrm{in} & v_\mathrm{w} & \omega_0 \end{bmatrix}^\mathrm{T}, \quad \boldsymbol{u} = \begin{bmatrix} M_\mathrm{g} & \beta_1 & \beta_2 & \beta_3 \end{bmatrix}^\mathrm{T}$$

$$\boldsymbol{d} = [\,], \quad \boldsymbol{y} = \begin{bmatrix} \ddot{x}_\mathrm{T} & \ddot{y}_\mathrm{T} & n_\mathrm{g} & \varphi \end{bmatrix}^\mathrm{T}$$

$$\boldsymbol{z} = \begin{bmatrix} M_\mathrm{a} & P_\mathrm{g} & M_{\mathrm{T}y} & M_{\mathrm{T}x} & M_\mathrm{DT} & F_\mathrm{T} \end{bmatrix}^\mathrm{T}, \quad \boldsymbol{p} = \begin{bmatrix} \delta'_\mathrm{v} & \delta'_\mathrm{h} \end{bmatrix}^\mathrm{T}$$

State Equations:

$$\dot{x}_1 = \frac{\varrho}{6}\frac{\pi R^2}{m_\mathrm{T}} \sum_{b=1}^{3} \Big(1 + \zeta r_\mathrm{n} \cos\psi_b\Big) C_\mathrm{T}(\lambda_b, u_{b+1}) v_b^2 - 2\zeta_{\mathrm{T}x} x_{12} x_1 - x_{12}^2 x_5$$

$$\dot{x}_2 = -\frac{\varrho}{6}\frac{\pi R^2}{m_\mathrm{T} r_\mathrm{t}} \sum_{b=1}^{3} \cos\psi_b \, C_\mathrm{M}(\lambda_b, u_{b+1}) v_b^2 - \frac{\zeta i_\mathrm{gb}}{m_\mathrm{T}} u_1 - 2\zeta_{\mathrm{T}y} x_{12} x_2 - x_{12}^2 x_6$$

$$\dot{x}_3 = \frac{1}{\Theta_\mathrm{g} i_\mathrm{gb}^2}(b_\varphi x_4 + k_\varphi x_8 - i_\mathrm{gb} u_1)$$

$$\dot{x}_4 = \frac{\varrho}{6}\frac{\pi R^3}{\Theta_\mathrm{r}} \sum_{b=1}^{3} C_\mathrm{M}(\lambda_b, u_{b+1}) v_b^2 + \frac{u_1}{\Theta_\mathrm{g} i_\mathrm{gb}} - \frac{\Theta_\mathrm{r} + \Theta_\mathrm{g} i_\mathrm{gb}^2}{\Theta_\mathrm{r} \Theta_\mathrm{g} i_\mathrm{gb}^2}\Big(b_\varphi x_4 + k_\varphi x_8\Big)$$

$$\dot{x}_5 = x_1\,, \quad \dot{x}_6 = x_2\,, \quad \dot{x}_7 = x_3\,, \quad \dot{x}_8 = x_4$$

$$\dot{x}_9 = T_\mathrm{w}^{-1}(x_{10} - x_9)$$

$$\dot{x}_{10} = \frac{\pi}{2L} x_9 (\chi(\psi_b) x_{11} - x_{10})$$

$$\dot{x}_{11} = 0\,, \quad \dot{x}_{12} = 0$$

Output Equations:

$$y_1 = \frac{\varrho}{6}\frac{\pi R^2}{m_\mathrm{T}} \sum_{b=1}^{3} \Big(1 + \zeta r_\mathrm{n} \cos\psi_b\Big) C_\mathrm{T}(\lambda_b, u_{b+1}) v_b^2 - 2\zeta_{\mathrm{T}x} x_{12} x_1 - x_{12}^2 x_5$$

$$y_2 = \frac{\varrho}{6}\frac{\pi R^2}{m_\mathrm{T} r_\mathrm{t}} \sum_{b=1}^{3} \cos\psi_b \, C_\mathrm{M}(\lambda_b, u_{b+1}) v_b^2 + \frac{\zeta i_\mathrm{gb}}{m_\mathrm{T}} u_1 + 2\zeta_{\mathrm{T}y} x_{12} x_2 + x_{12}^2 x_6$$

$$y_3 = 30\,\pi^{-1} i_\mathrm{gb} x_1$$

$$y_4 = x_7 + x_8$$

Performance Eqs.:

$$z_1 = \frac{\varrho}{2}\frac{\pi R^3}{3} \sum_{b=1}^{3} C_\mathrm{M}(\lambda_b, u_{b+1}) v_b^2\,, \quad z_2 = \eta_\mathrm{g} i_\mathrm{gb} x_1 u_1$$

$$z_3 = \Big(2\zeta_{\mathrm{T}x} x_{12} x_1 + x_{12}^2 x_5\Big) m_\mathrm{T} H\,, \quad z_4 = \Big(2\zeta_{\mathrm{T}y} x_{12} x_2 + x_{12}^2 x_6\Big) m_\mathrm{T} H$$

$$z_5 = b_\varphi x_4 + k_\varphi x_8\,, \quad z_6 = \frac{\varrho}{2}\frac{\pi R^2}{3} \sum_{b=1}^{3} C_\mathrm{T}(\lambda_b, u_{b+1}) v_b^2$$

Definitions:

$$\lambda_b = \frac{(x_3 + x_4) R}{v_b}\,, \quad x_{12}^2 = \frac{k_{\mathrm{T}x}}{m_\mathrm{T}}\,, \quad 2\zeta_{\mathrm{T}x} x_{12} = \frac{b_{\mathrm{T}x}}{m_\mathrm{T}}\,, \quad i_\mathrm{gb} = \frac{n_\mathrm{g}}{n_\mathrm{r}} = \frac{2\pi}{60}\frac{n_\mathrm{g}}{\dot{\varphi}}$$

$$v_b = x_{10} + \frac{r_\mathrm{B}}{2R}\Big(p_1 \cos\psi_b + p_2 \sin\psi_b\Big) - \Big(1 + \zeta r_\mathrm{B} \cos\psi_b\Big) x_1$$

$$\psi_b = x_7 + x_8 + \frac{2\pi}{3}(b-1)$$

3.4.6 Model Parameters for the NREL Reference Turbine

Most of the research is and was conducted with the 5 MW reference turbine defined by the NREL [95]. Since not all of the needed parameters are directly provided by the report, the following Tab. 3.3 summarizes them. Fig. 3.7 shows the corresponding aerodynamic coefficients as contour plots (cf. p. 53).

Table 3.3: List of the model and turbine parameters

Symbol	Value	Unit	Symbol	Value	Unit
$k_{\mathrm{T}x}$	1.92×10^6	N/m	$M_{\mathrm{a},0}$	4180	kNm
$b_{\mathrm{T}x}$	18.6×10^3	Ns/m	ϱ	1.22	$\mathrm{kg/m^3}$
$k_{\mathrm{T}y}$	1.92×10^6	N/m	k_φ	867×10^6	Nm/rad
$b_{\mathrm{T}y}$	9.32×10^3	Ns/m	b_φ	6.22×10^6	Nms/rad
m_{T}	450×10^3	kg	Θ	40.5×10^6	$\mathrm{kg\,m^2}$
R	63	m	Θ_{r}	35.5×10^6	$\mathrm{kg\,m^2}$
r_{B}	47.3	m	Θ_{g}	534	$\mathrm{kg\,m^2}$
r_{n}	31.5	m	η_{g}	0.944	-
r_{t}	31.5	m	ω_0	2.07	1/s
H	87.6	m	λ^*	7.6	-
i_{gb}	97	-	C_{P}^*	0.475	-
T_{w}	0.1	s	$\zeta_{\mathrm{T}x}$	0.01	-
$n_{\mathrm{r},0}$	12.1	rpm	$\zeta_{\mathrm{T}y}$	0.005	-
$n_{\mathrm{g},0}$	1173	rpm	μ_{T}	0.05	-
$P_{\mathrm{g},0}$	5000	kW	σ_{T}	25	-
$M_{\mathrm{g},0}$	43.1	kNm	ζ	0.017	1/m

3.5 Summary

This chapter started with a general introduction to wind energy and to wind turbines, in particular. The physical basics on wind energy conversion were explained concisely in order to establish a fundamental understanding. This was followed by a technological overview of modern wind turbines from a control engineer's perspective (which included the main components, actuators and measurement equipment).

Hereafter, different control-oriented models were presented for different purposes such as for wind turbine simulation, control and state estimation tasks. That needs at least a wind field sub-model, an aerodynamic sub-model and a mechanical sub-model to characterize the wind turbine dynamics adequately. The first model describes the ambient wind field as the major source of disturbance and uncertainty (represented by the effective wind speed and the shears). Secondly, this wind field interacts with the rotor blades which is modelled by simplified fluid dynamic principles. Thirdly, the resulting aerodynamic forces and torques excite (in turn) forced mechanical vibrations of the structural model. Bringing together

these three effects from the distinct sub-models, yielded several state space representations with different focus and model granularity.

Subsequently, these developed design models are applied to establish a novel and automated design methodology for wind turbine estimators. This methodology is founded upon the appropriate and well-parametrized models in order to manage the transfer from simple control-oriented models to more advanced and realistic simulators.

4 Optimal Filter Parameter Design

The favourable choice of the Kalman filter's parameters often appears to be a mystery to non-experts. Indeed, the manual design procedure has sometimes very little to do with a systematic design approach. Yet, the estimation results for this intuitive design are often acceptable but may remain behind the theoretically possible results. The more elaborated way to achieve the best filter results is pursued by solving a well-defined optimization problem numerically. This is also of essential importance for the use in highly automated frameworks to determine the free design parameters automatically with little user interaction.

4.1 Introduction

This chapter deals with the optimal and automated design of the free filter parameters. Free parameters are those which cannot be determined from a priori knowledge only and affect the estimation performance to a critical degree. This is a well-known fact to every expert who is confronted to provide a filter design that meets the expected estimation performance. Therefore measures must be taken in order to make the filter robust against the uncertainty from process and measurement noise [2].

In general, filter design is often referred to as the process of selecting the free filter parameters such that the estimation performance is optimized with respect to a dedicated and well chosen objective function. The objective may be a well-defined mathematical criterion or an expert opinion gained from assessing the estimation results. The latter approach is also denoted as *tuning process* when parameters are manually designed by expert knowledge. This is often considered as best practice in Kalman filter design since the importance is readily shifted from one state to another.

Yet, this manual process becomes very time-consuming and cumbersome when done on a trial and error basis [55]. This is especially true for an increasing number of estimated states (thus also more free parameters) which makes it even harder to achieve satisfying results in limited time. Additionally, the manual design is not a very practical whenever the design process must be repeated frequently, for instance due to a need for individual customizing. Such customization is indicated e.g. because of changing from one turbine design to the next turbine design or depending on the wind site. Therefore, an automated filter design based on numerical optimization is desirable to ease the application, to reduce time and effort as well as to optimize the filter performance [116] (p. 6). Optimization is also termed as mathematical programming which addresses the iterative search for an optimal

solution of a mathematically defined problem [142] (p. 8/9). Nevertheless, there are a couple of requirements that need to be fulfilled before the optimal filter design is found eventually:

1. The model structure and model parameters must be selected.
2. The implementation of the filter algorithm must be realized.
3. The performance criteria must be selected to represent the objectives adequately.
4. The generic optimization problem must be formulated.
5. The estimation problem must be defined.
6. The test scenario must be selected and simulation data must be generated.
7. The specific optimization problem must be solved.
8. The overall performance of the found optimal design must be assessed.

The above Items (1) and (2) have already been treated comprehensively in Sect. 2.3 and 3.4.[23] Moreover, open source implementations of the SPKF are available for free.[24] With regard to Item (3), basic considerations were introduced in Sect. 2.4. More details are provided in the subsequent Sect. 4.2.1. The Items (4) to (7) are mainly addressed in the present chapter. Item (4) is treated in detail in Sect. 4.2.2 where the proposed optimization problem is introduced with two different objective functions. The Item (5) and (6) are briefly highlighted in Sect. 4.3.1 (p. 89). Item (7) is discussed in Sect. 4.2.4 and the results in Sect. 4.3.3.

Finally, Item (8) constitutes the most relevant part prior to application. This means that it must be made sure that the found filter design works in every relevant test scenario. Therefore, this performance test is treated extensively in Chap. 5 (p. 105 ff) where several different test scenarios are applied to the optimally designed filters.

In summary, the main goals of this chapter are

- to develop a novel design methodology for optimal and automated filter parameter design (in order to avoid the intuitive tuning process to a large degree),
- to formulate a generic optimization problem to KF design, and
- to apply this design methodology to a relevant state estimation problem with an arbitrary initial filter design.

[23] The used model parameters for the NREL reference turbine are listed in Sect. 3.3 (p. 78).

[24] see for instance `https://haranarasaratnam.com/software.html` [accessed 2020-01-13] for a rudimentary implementation of the cubature Kalman filter which must be further improved for industrial application.

In order to realize the above formulated goals, the remainder of this chapter is organized as follows: Sect. 4.2 defines the generic optimization problem and proposes suitable objective functions for that purpose. Then, Section 4.3 discusses the optimal and automated design for the static filter parameters. Sect. 4.4 applies a similar methodology to the slave Kalman filter and presents illustrative simulation data for the time-varying noise case.

4.2 Generic Optimization Problem

A precondition for an optimization-based filter design is the definition of a suitable objective function. This is likewise critical for optimal control (like NMPC) as well as optimal state estimation. Without an appropriate criterion, every automated filter design is doomed to failure. In this section, two different cost functions are presented which both include normalized values of the estimation error $\boldsymbol{e}_k$ and the innovation $\boldsymbol{v}_k$. The difference is that the first cost function J_{WS} uses constant and predefined normalization weights while the second one J_{NS} uses the filter covariance matrices as weights (cf. p. 84 ff).

4.2.1 Performance Measures

There are two important quantities when it comes to filter performance testing. These are the estimation error $\boldsymbol{e}_k$ (EE) and the measurement innovation $\boldsymbol{v}_k$ which are defined by

$$\boldsymbol{e}_k = \boldsymbol{x}_k - \hat{\boldsymbol{x}}_k^+ \quad \text{and} \quad \boldsymbol{v}_k = \boldsymbol{y}_k - \hat{\boldsymbol{y}}_k \ . \tag{4.1}$$

The first is often only available in simulation or rarely with advanced sensor equipment when $\boldsymbol{x}_k$ is known. The second quantity is always available in reality because $\boldsymbol{y}_k$ is measured. Based on the Eqs. (4.1), the weighted estimation error squared (WEES) and the weighted innovation squared (WIS) are proposed as

$$\varepsilon_{\mathrm{ws},k} = \boldsymbol{e}_k^{\mathrm{T}}\, \mathbf{W}_x\, \boldsymbol{e}_k \tag{4.2}$$

$$\nu_{\mathrm{ws},k} = \boldsymbol{v}_k^{\mathrm{T}}\, \mathbf{W}_y\, \boldsymbol{v}_k \tag{4.3}$$

where the matrices $\mathbf{W}_x \succeq 0$ and $\mathbf{W}_y \succeq 0$ serve as weights to penalize the EE and the innovation one by one. Since sometimes not only the absolute values $\boldsymbol{e}_k$ and $\boldsymbol{v}_k$ are of interest but also its relative changes, the Eqs. (4.2) and (4.3) are enhanced such that

$$\varepsilon_{\mathrm{ws},k} = \boldsymbol{e}_k^{\mathrm{T}}\, \mathbf{W}_x\, \boldsymbol{e}_k + \dot{\boldsymbol{e}}_k^{\mathrm{T}}\, \mathbf{W}_{\dot{x}}\, \dot{\boldsymbol{e}}_k \tag{4.4}$$

$$\nu_{\mathrm{ws},k} = \boldsymbol{v}_k^{\mathrm{T}}\, \mathbf{W}_y\, \boldsymbol{v}_k + \dot{\boldsymbol{v}}_k^{\mathrm{T}}\, \mathbf{W}_{\dot{y}}\, \dot{\boldsymbol{v}}_k \tag{4.5}$$

holds. Thereby, also the errors in the first time-derivatives are considered in the objective function with the weighting matrices $\mathbf{W}_{\dot{x}} \succeq 0$ and $\mathbf{W}_{\dot{y}} \succeq 0$ which allows for more direct control over the design process. Based on Eqs. (4.4) and (4.5), the sample averages over a batch size M are then defined as

$$\bar{\varepsilon}_{\mathrm{ws}} = \frac{1}{M} \sum_{k=1}^{M} \varepsilon_{\mathrm{ws},k} \quad \text{and} \quad \bar{\nu}_{\mathrm{ws}} = \frac{1}{M} \sum_{k=1}^{M} \nu_{\mathrm{ws},k} \tag{4.6}$$

where the average weighted EE squared is $\bar{\varepsilon}_{\mathrm{ws}}$ and the sum of weighted innovation squared is $\bar{\nu}_{\mathrm{ws}}$. For sake of simplicity, all weighting matrices $\mathbf{W}_x$, $\mathbf{W}_{\dot{x}}$, $\mathbf{W}_y$ and $\mathbf{W}_{\dot{y}}$ are assumed to be diagonal which yields altogether $2n_x + 2n_y$ design parameters. This gives the filter designer extensive possibilities to control the design process. Unfortunately, the deliberate choice of the diagonal elements does not necessarily provide a consistent estimator by design, as discussed in the following.

Consistency is given when the empirical covariances matrices (drawn from the EE and the innovation) are in line with the filter computed covariance matrices. To put it simple, the estimator's property of consistency ensures that the filter algorithm is always aware of how reliable its estimates actually are [40]. In order to check the filter consistency, the normalized estimation error squared $\varepsilon_{\mathrm{NS},k}$ (NEES) and the normalized innovation squared $\nu_{\mathrm{NS},k}$ (NIS) are introduced. In accordance with [9] (p. 165 / p. 236), the definitions read as follows:

$$\varepsilon_{\mathrm{NS},k} = \boldsymbol{e}_k^{\mathrm{T}}\, \mathbf{M}_k^{+} \boldsymbol{e}_k \quad \Rightarrow \quad \varepsilon_{\mathrm{NS},k} \sim \chi^2_{n_x} \tag{4.7}$$

$$\nu_{\mathrm{NS},k} = \boldsymbol{v}_k^{\mathrm{T}}\, \mathbf{M}_k^{yy} \boldsymbol{v}_k \quad \Rightarrow \quad \nu_{\mathrm{NS},k} \sim \chi^2_{n_y}\ . \tag{4.8}$$

Both quantities are χ^2-distributed with n_x or n_y degrees of freedom. $\mathbf{M}_k^{+}$ and $\mathbf{M}_k^{yy}$ are the matrix inverses of $\mathbf{P}_k^{+}$ and $\mathbf{P}_k^{yy}$, respectively. Thus, the weighting matrices from Eqs. (4.2) and (4.3) are replaced by the inverses of the filter computed covariances. As a matter of fact, this yields two performance measures (4.7) and (4.8) which are completely free of design parameter choices.

Further on, it is well known that filter consistency implies in principle that

$$E\{\varepsilon_{\mathrm{NS},k}\} = n_x \quad \text{and} \quad E\{\nu_{\mathrm{NS},k}\} = n_y \tag{4.9}$$

must hold [9] (p. 59). In order to assess consistency, the Eqs. (4.9) can be used readily in a statistical sense – evaluating the empirical means of the NEES and the NIS – as

$$\bar{\varepsilon}_{\mathrm{NS}} = \frac{1}{n_x M} \sum_{k=1}^{M} \varepsilon_{\mathrm{NS},k} \quad \text{and} \quad \bar{\nu}_{\mathrm{NS}} = \frac{1}{n_y M} \sum_{k=1}^{M} \nu_{\mathrm{NS},k} \tag{4.10}$$

which represent the normalized sample means (obtained for altogether M filter sample steps). Then an estimator is said to be consistent when

$$\bar{\varepsilon}_{\mathrm{NS}} \approx 1 \quad \text{and} \quad \bar{\nu}_{\mathrm{NS}} \approx 1 \tag{4.11}$$

holds. These criteria can be assessed both for filter design (see below) and also online performance assessment (at least for the innovation).

4.2.2 Objective Functions and Optimization Problem

Based on the above measures, two different objective functions to assess the filter's performance are proposed which both rely on the normalized state and measurement errors. The first one is based on Eqs. (4.4), (4.5) and (4.6). It is defined as

$$J_{\mathrm{WS}}(\boldsymbol{z}) = \bar{\varepsilon}_{\mathrm{ws}}(\boldsymbol{z}) + \bar{\nu}_{\mathrm{ws}}(\boldsymbol{z}) \tag{4.12}$$

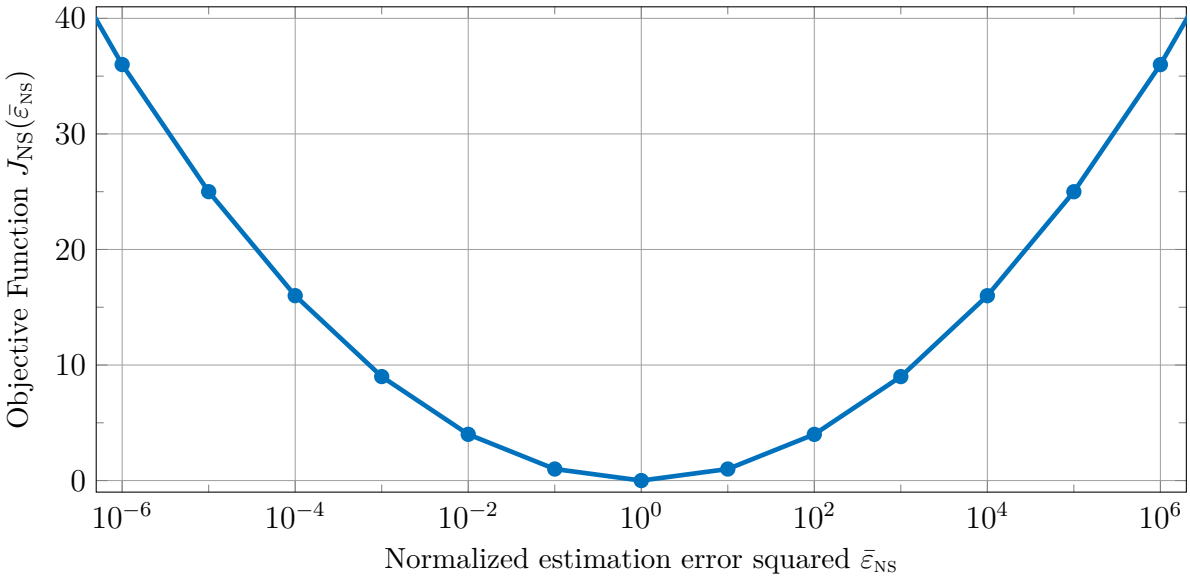

Figure 4.1: The objective function $J_{\mathrm{NS}}(\bar{\varepsilon}_{\mathrm{NS}})$ in dependency of the NEES (evaluated for $w = 1$)

wherein $\boldsymbol{z}$ contains the free design parameters (diagonal elements of process and measurement noise covariances) to be optimized. This first objective function $J_{\mathrm{WS}}(\boldsymbol{z})$ constitutes a heuristic measure that penalizes directly $\boldsymbol{e}_k$ and $\boldsymbol{v}_k$ in the mean-squared error (MSE) sense. Despite being very practical, the drawback is that $J_{\mathrm{WS}}(\boldsymbol{z})$ does not necessarily yield a consistent estimator but *just* an estimator that minimizes the MSE. Yet, the main advantage is that the practitioner has extensive control over the state estimation errors as often desired. Thus, adjustment possibilities are provided with the choice of $\mathbf{W}_x$ and $\mathbf{W}_y$ (and also $\mathbf{W}_{\dot{x}}$ and $\mathbf{W}_{\dot{y}}$) which makes the design procedure very potent (cf. p. 83).

The second cost function uses the error covariance matrices $\mathbf{P}_k^{+}(\boldsymbol{z})$ and $\mathbf{P}_k^{yy}(\boldsymbol{z})$ as weights instead.[25] It applies the performance measures from Eqs. (4.10) and is proposed as

$$J_{\mathrm{NS}}(\boldsymbol{z}) = w \log_{10}^2\Big(\bar{\varepsilon}_{\mathrm{NS}}(\boldsymbol{z})\Big) + (1-w) \log_{10}^2\Big(\bar{\nu}_{\mathrm{NS}}(\boldsymbol{z})\Big) \tag{4.13}$$

where $0 \leq w \leq 1$ is a weighting factor used to shift the focus from the NEES to the NIS (et v.v.), depending on the situation if or if not the true states are known.

This objective function $J_{\mathrm{NS}}(\boldsymbol{z})$ is based on the *consistency thinking* above, cf. Eq. (4.9). The logarithmic function in Eq. (4.13) is chosen intentionally in order to match the requirement, as stated in Eq. (4.11), that the optimum J^* is zero in the ideal case, since $\log_{10}(1) = 0$ applies. The square rule enforces the desired convexity. This objective function Eq. (4.13) has three advantages: First, it ensures that pessimism and optimism in filter tuning [40] are weighted equally (cf. Fig. 4.1). Secondly, there are no tuning parameters since the weights are predefined by the filter covariance matrices in Eqs. (4.7) and (4.8). Thirdly, Eq. (4.13) provides a statistically consistent estimator when the optimum J^* is found.

[25] This eliminates the need for a manual choice of the weights as it is required in Eq. (4.12).

On the downside, the main disadvantage is that the convergence to the optimal solution is questionable if the number of optimization variables is not restricted. Loosely speaking, the filter consistency depends on the ratio of the noise covariance matrices [35] (p. 154) and thus one of these two quantities needs to be set prior to optimization.

Based on the previously mentioned considerations, the generic optimization problem (OP) for the automated filter design is constituted as follows:

$$\begin{aligned} \min_{\boldsymbol{z}} \;& J(\boldsymbol{z}) \\ \text{s.t.} \;& [\hat{\boldsymbol{x}}_i^+(\boldsymbol{z}), \hat{\boldsymbol{y}}_i(\boldsymbol{z})] = \mathrm{F}(\boldsymbol{z}, \boldsymbol{u}_i, \boldsymbol{y}_i) \;:\; \forall i \in [1, 2, \ldots, M] \\ & \underline{\boldsymbol{z}} \leq \boldsymbol{z} \leq \bar{\boldsymbol{z}} \end{aligned} \tag{4.14}$$

Therein, $J(\boldsymbol{z})$ can be chosen either Eq. (4.12) or (4.13). The advantages and disadvantages of both are discussed below. The function $\mathrm{F}(\cdot)$ represents the chosen filter algorithm (e.g. a SPKF-type with or without adaptation) which supplies the corrected state estimate $\hat{\boldsymbol{x}}_i^+$ and the predicted output $\hat{\boldsymbol{y}}_i$ in every sampling step. The test scenario (including the wind scenario, the controller, the measurement configuration, the simulator, etc.) is determined by the trajectories of the control inputs $\boldsymbol{u}_i$ and measurement outputs $\boldsymbol{y}_i$. The trajectory of the true states $\boldsymbol{x}_i$ is required to be known if the NEES or the WEES is involved/needed in the cost function $J(\boldsymbol{z})$.

The optimization variables $\boldsymbol{z}$ contain the released design parameters and are limited by the lower bounds $\underline{\boldsymbol{z}}$ and the upper bounds $\bar{\boldsymbol{z}}$. In particular, the lower bounds $\underline{\boldsymbol{z}}$ are mandatory in order to enforce/guarantee that the covariance matrices $\mathbf{R}_k$ and $\mathbf{Q}_k$ (for master filter design) or $\mathbf{Q}_{\mathrm{s},k}$ and $\mathbf{R}_{\mathrm{s},k}$ (for slave filter design) remain positive (semi-)definite.

In summary, both proposed cost functions have the potential to circumvent the intuitive design approach completely (or at least to enhance it significantly). Rather than choosing the noise covariances directly, the automated design reduces to choosing weights in a well-defined optimization problem.

4.2.3 Choice of the Initial Values and Limits

Despite the choice of a convex objective function, the established optimization problem in Eq. (4.14) itself, is potentially not convex at all. Due to the nonlinear filter algorithm and a nonlinear internal system model, the constraints in Eq. (4.14) are likely to be non-convex functions. Hence, this necessitates the diligent choice of the initial optimization variables $\boldsymbol{z}_0$ since a global optimum cannot be guaranteed in general.

For the initial design (and also the optimal design later on) only the diagonal elements of the noise covariances are released for design,

$$\mathbf{Q}_0^0 = \mathrm{diag}\left\{ \begin{bmatrix} q_{11}^0 & q_{22}^0 & \cdots & q_{ii}^0 & \cdots & q_{n_x n_x}^0 \end{bmatrix}^{\mathrm{T}} \right\} \tag{4.15}$$

$$\mathbf{R}_0^0 = \mathrm{diag}\left\{ \begin{bmatrix} r_{11}^0 & r_{22}^0 & \cdots & r_{jj}^0 & \cdots & r_{n_y n_y}^0 \end{bmatrix}^{\mathrm{T}} \right\} \tag{4.16}$$

since these are well-known to have the largest effect on filter performance, cf. [160] (p. 903). Consequently, all off-diagonal elements are set to zero which is a reasonable (and advisable) assumption since often physically hard to interpret. In order to select the initial values q_{ii}^0 and r_{jj}^0 appropriately, a simple intuitive choice of parameters is made that involves only little problem-specific prior knowledge. That is, simple choices like unit or zero matrices are preferable which shall facilitate the automated design approach.

For sake of convenience, the initial filter parameters are equal for every scenario and chosen to be $q_{ii}^0 = 10^{-6}$ and $r_{jj}^0 = 1$. Practically, zeros in the diagonal elements of the process noise $\mathbf{Q}_k$ are avoided for numerical reasons and in order to introduce a certain amount of uncertainty into the filter in every observation direction, cf. Bar-Shalom et al. [9] (p. 482). The variances must only comply with $\mathbf{Q}_0 \succeq 0$ and $\mathbf{R}_0 \succ 0$ which is the prerequisite for the Kalman filter to function. Thus, these matrices must be positive (semi-)definite.

Besides the initial values, also the lower and upper bound constraints must be confined to the physically meaningful region so that the estimator does not diverge within the optimization procedure. The lower bound for process noise are set to $\underline{q}_{ii} = 10^{-9}$ while the upper bound is $\bar{q}_{ii} = 10^{0}$. The limits for the sensor noise covariances are chosen according to the actual sensor signal and the expectable uncertainty.

4.2.4 Numerical Solution

This section motivates briefly the approach to select the numerical solver for the SPKF design and the reasoning why interior-point methods (IPM) are employed.

In principle, executing a linear or nonlinear KF within an optimization problem (OP) can be problematic from numerical point of view since the filter may diverge from one iteration to the next. Such undesired behaviour may occur if the filter becomes unstable or if there is more than one solution to the estimation problem. It is hard to show that such an OP including nonlinear systems and filters is globally convex (which is often a necessary condition) and therefore to prove that a global optimum exist and can be reached.

On the other hand, the basic idea is to simplify and to automate the design process, and not to make it more complicated. Despite the theoretical possible problems, the practical experience often supports the application also to non-convex nonlinear problems when the decision variables (in this case the noise parameters) are bounded to a meaningful region.

As a matter of fact, the key aspects, that are considered as criteria for the selection of the numerical solver, are summarized as follows:

- The solver deals readily with nonlinear optimization problems due to the nonlinear wind turbine system and the filter algorithm.
- The system constraints are directly integrable and the solver enforces these constraints at every iteration step to avoid filter divergence.

- Existing solvers are preferable over individual realization since than the numerical implementation is more reliable since tested and applied in a greater community.
- The solver type itself is algorithmically mature.
- The software is in principle available to anyone and is regularly maintained.

Hence, the algorithm must be capable to manage a nonlinear constrained optimization problem (potentially non-convex) and also provide the necessary maturity (software and functionality).

Since the tool chain is completely developed in Matlab/Simulink, the idea suggests itself to use existing built-in routines. In particular, a tailored algorithm for constrained nonlinear optimization is the built-in function *fmincon* which provides different solver types such as the IPM, active-set method (ASM) and sequential quadratic programming (SQP) methods [142] (p. 529 ff). The IPM types are conceivable as general purpose solvers for optimization problems, cf. [159, 222, 221]. These belong to the newest class of algorithms to solve nonlinear optimization problems [142] (p. 424). On the contrary, the SQP show their strength when dealing with problems with strong nonlinear constraints [142] (p. 529).

The three types were investigated in a pre-study in order to assess their performance in an environment with artificial sensor noise. The results are roughly summarized as follows:

- The IPM and the SQP show in principle consistent results with respect to the optimal solution $\boldsymbol{z}^*$ and J^* for low and moderate sensor noise levels.
- With increasing artificial sensor noise level, the IPM appears to be more reliable to provide consistent results than the SQP.
- The ASM provide most of the times inconsistent results compared to the IPM and SQP results. The obvious reason is that the optimization terminates relatively early and apparently prematurely though the numerical reasoning is not clear.
- A comparison regarding the number of iterations revealed that the IPM show often less iterations compared to the same scenarios with SQP.

Based on the above points, first ASM are excluded from the further investigation since no useful results can be obtained, and secondly IPM are favoured over the SQP due to their more robust results and less iterations needed.

In summary, the IPM provide the functionality needed for the automated design and show a similar performance with SQP. The various simulation results and experience gained from this research by the author support the decision to focus on Matlab's realization of the IPM in the following. The detailed study of the numerically properties is out-of-scope though. For further reading, the textbook by Nocedal & Wright [142] is recommended which establishes vividly the fundamental understanding for unconstrained and constrained nonlinear optimization.

4.3 Design of the Classical Filter

The static filter design must be conducted for two reasons. First, the classical SPKF needs a proper initialization of its noise covariances $\mathbf{Q}_k$ and $\mathbf{R}_k$ for $k = 0$. Secondly, the adaptive filter needs initial values for the adapted and also the fixed parameters.

The importance of a *correct* configuration of these process and measurement noises is highlighted by several authors including Stengel [203], Maybeck [128] and Busse [35] (p. 152 ff). The automated design is conceived in this regard as finding the *correct* filter parameter configuration with little manual interaction and preferably independent of the estimation problem and the test scenario configuration. In the following, the principle approach to find the optimal design is exerted for the joint wind and state estimation problem.

4.3.1 Definition of the Estimation Problem

The joint estimation problem considered hereafter for the classical filter design denotes:

> Estimate the nacelle fore-aft dynamics $\dot{x}_\mathrm{T}$ and x_T as well as the unknown disturbance input v_w together with a monolithic CKF! Consider for this purpose a moderate sensor noise environment using realistic data $\boldsymbol{u}_k$ and $\boldsymbol{y}_k$ (cf. Fig. 4.2) as filter inputs and $\boldsymbol{x}_k$ for validation purposes from the FAST simulator.

In order to fulfil this estimation task, the Model 3.4 suits very well and is therefore used as the simplified filter model internally. As a reminder, the augmented state vector writes

$$\boldsymbol{x} = \begin{bmatrix} \dot{\varphi}_\mathrm{g} & \dot{x}_\mathrm{T} & x_\mathrm{T} & v_\mathrm{w} \end{bmatrix}^\mathrm{T} \tag{4.17}$$

where $\dot{\varphi}_\mathrm{g}$ is the generator angular speed, $\dot{x}_\mathrm{T}$ and x_T are the nacelle velocity and position, respectively, and v_w is the rotor-effective wind speed. Moreover, the available information for the estimator are the two control inputs and the two sensor outputs that read

$$\boldsymbol{u} = \begin{bmatrix} M_\mathrm{g} & \beta_\mathrm{c} \end{bmatrix}^\mathrm{T} \quad \text{and} \quad \boldsymbol{y} = \begin{bmatrix} n_\mathrm{g} & \ddot{x}_\mathrm{T} \end{bmatrix}^\mathrm{T}. \tag{4.18}$$

To demonstrate the functionality of the proposed approach for this model, an illustrative test scenario is required which is introduced in Fig. 4.2. Therein, the simulation data for a turbulent wind field with an average wind speed $v_\mathrm{m} = 12\,\mathrm{m/s}$ is shown including also the measurement errors Δn_g and $\Delta \ddot{x}_\mathrm{T}$. Moreover, the curve of the generator power P_g indicates that both, the PLR and FLR, are included in the scenario alike which is desirable to achieve an optimal design that works in the whole operating range.

In addition, the turbulence of such a test scenario must not be too small since the noise covariances shall to be identified. Therefore, a sufficient excitation is critical for the success of the optimization (as it is well-known for parameter identification).

The definition of the considered noise levels is provided in Tab. 4.1 which includes five steps from (effectively) zero noise to maximum noise.

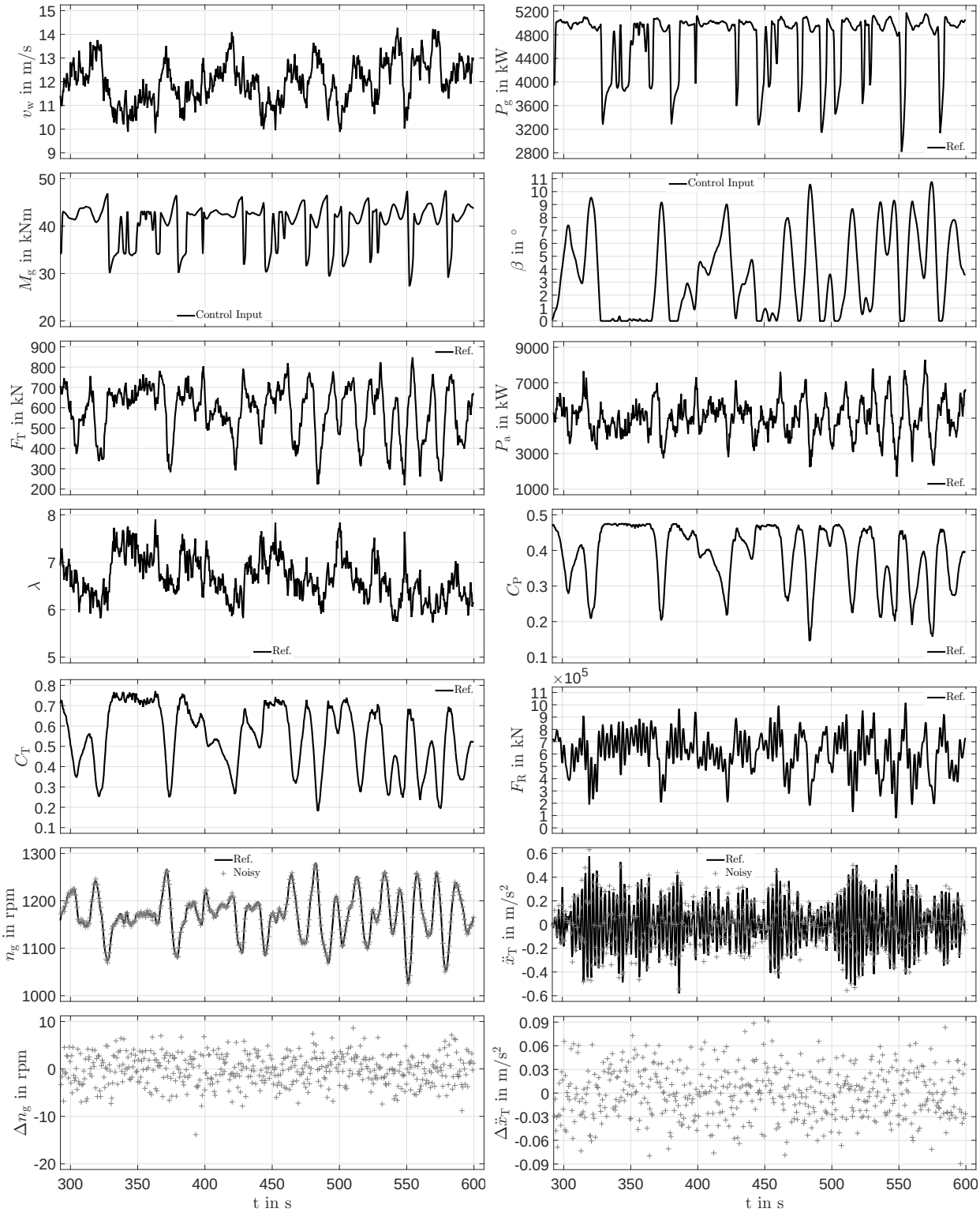

Figure 4.2: This test scenario is used to conduct the optimal static design procedure. The average wind speed is 12 m/s and the sensor noises are chosen to be zero-mean with the noise covariances $\mathrm{Var}\{n_g\} = 10\,(\mathrm{rpm})^2$ and $\mathrm{Var}\{\ddot{x}_T\} = 10^{-3}\,(\mathrm{m/s^2})^2$ (hence a medium sensor noise environment, cf. Tab. 4.1)

Table 4.1: Definition of sensor specific noise levels

		zero	low	medium	high	max.
$r_{11} = \mathrm{Var}\{n_\mathrm{g}\}$	in $(\mathrm{rpm})^2$	10^{-1}	10^{+0}	10^{+1}	10^{+2}	10^{+3}
$r_{22} = \mathrm{Var}\{\ddot{x}_\mathrm{T}\}$	in $(\mathrm{m/s^2})^2$	10^{-5}	10^{-4}	10^{-3}	10^{-2}	10^{-1}

Table 4.2: Selected constraints and initial values for the decision variables $\boldsymbol{z}$

decision variable	q_{11}	q_{22}	q_{33}	q_{44}	r_{11}	r_{22}
physical unit	in $(1/\mathrm{s})^2$	in $(\mathrm{m/s})^2$	in $(\mathrm{m})^2$	in $(\mathrm{m/s})^2$	in $(\mathrm{rpm})^2$	in $(\mathrm{m/s^2})^2$
upper bound	10^{0}	10^{0}	10^{0}	10^{0}	10^{3}	10^{1}
initial value	10^{-6}	10^{-6}	10^{-6}	10^{-6}	10^{0}	10^{0}
lower bound	10^{-9}	10^{-9}	10^{-9}	10^{-9}	10^{-3}	10^{-5}

4.3.2 Choice of the Decision Variables and Weights

Since there are $n_x = 4$ model states and $n_y = 2$ sensor outputs, in total $n_z = n_x + n_y = 6$ process and measurement noise variables $\boldsymbol{z} \in \mathbb{R}^6$ must be designed appropriately prior to application. For sake of convenience, the initial values are set equal for every scenario. The values are set in accordance with Sect. 4.2.3 (p. 86) to be

$$\mathbf{Q}_0^0 = \mathrm{diag}\Big\{\big[q_{11}^0\ q_{22}^0\ q_{33}^0\ q_{44}^0\big]^\mathrm{T}\Big\} = \mathrm{diag}\Big\{10^{-6}\big[1\ 1\ 1\ 1\big]^\mathrm{T}\Big\} \tag{4.19}$$

$$\mathbf{R}_0^0 = \mathrm{diag}\Big\{\big[r_{11}^0\ r_{22}^0\big]^\mathrm{T}\Big\} \qquad = \mathrm{diag}\Big\{\big[1\ 1\big]^\mathrm{T}\Big\} \tag{4.20}$$

wherein simple diagonal matrices are used to ease the automated design (calling for little prior knowledge).

WEES- and WIS-based Cost Function

For the cost function $J_\mathrm{WS}(\boldsymbol{z})$ from Eq. (4.12), the initial decision variables are chosen as

$$\boldsymbol{z}_\mathrm{WS}^0 = \big[q_{11}^0\ q_{22}^0\ q_{33}^0\ q_{44}^0\ r_{11}^0\ r_{22}^0\big]^\mathrm{T} \tag{4.21}$$

and thus all noise variables are optimized simultaneously. The upper and lower limits for these decision variables are gathered in Tab. 4.2 which comes from expert process knowledge on the allowable variations of the physical states. In particular the measurement noise covariances need to be restricted to physical meaningful values (which are defined as a region $[10^{-3}\ 10^{3}] \cdot r_{jj}^\mathrm{n}$ around a nominal value). The weighting matrices needed for $J_\mathrm{WS}(\boldsymbol{z})$ are selected to be

$$\mathbf{W}_x = \mathrm{diag}\Big\{\big[1\ 1\ 1\ 1\big]^\mathrm{T}\Big\}\ , \quad \mathbf{W}_y = \mathrm{diag}\Big\{\big[10^{-3}\ 1\big]^\mathrm{T}\Big\} \tag{4.22}$$

$$\mathbf{W}_{\dot{x}} = \mathrm{diag}\Big\{\big[1\ 1\ 1\ 1\big]^\mathrm{T}\Big\}\ , \quad \mathbf{W}_{\dot{y}} = \mathrm{diag}\Big\{\big[10^{-3}\ 1\big]^\mathrm{T}\Big\} \tag{4.23}$$

Table 4.3: Initial values of the noise covariances for process noise optimization

noise variable physical unit	q_{11}^0 in $(1/\mathrm{s})^2$	q_{22}^0 in $(\mathrm{m/s})^2$	q_{33}^0 in $(\mathrm{m})^2$	q_{44}^0 in $(\mathrm{m/s})^2$	r_{11} in $(\mathrm{rpm})^2$	r_{22} in $(\mathrm{m/s^2})^2$
case 2b → very optimistic	10^{-6}	10^{-6}	10^{-6}	10^{-6}	10^{-1}	10^{-5}
case 1b → optimistic	10^{-6}	10^{-6}	10^{-6}	10^{-6}	10^{0}	10^{-4}
case 0 → correct	10^{-6}	10^{-6}	10^{-6}	10^{-6}	10^{1}	10^{-3}
case 1a → pessimistic	10^{-6}	10^{-6}	10^{-6}	10^{-6}	10^{2}	10^{-2}
case 2a → very pessimistic	10^{-6}	10^{-6}	10^{-6}	10^{-6}	10^{3}	10^{-1}

which comes from expert's intuitive knowledge for normalized state variables (though a fairly simple choice).

NEES- and NIS-based Cost Function

Contrary, the cost function $J_{\mathrm{NS}}(\boldsymbol{z})$ in Eq. (4.13) is only parametrized by the weight w which is set to 0.5. The reason simply is that ground truth about the states is available from simulation and also measurements shall be included in the cost function equally[26]. As mentioned earlier on p. 86, not all noise parameters are released for optimization. Since the adaptive filter in Sect. 4.4 addresses the problem of incorrect sensor noise parameters, here only the process noise variables are optimized and thus

$$\boldsymbol{z}_{\mathrm{NS}}^0 = \left[q_{11}^0 \;\; q_{22}^0 \;\; q_{33}^0 \;\; q_{44}^0\right]^{\mathrm{T}} \tag{4.24}$$

is chosen. The remaining two parameters are set to the pre-defined sensor noise levels

$$r_{11} = 10\,(\mathrm{rpm})^2 \quad \text{and} \quad r_{22} = 10^{-3}\,(\mathrm{m/s^2})^2 \tag{4.25}$$

of the test scenario, cf. Fig. 4.2, and are considered as the *correct* noise parameters (true values). In order to test the influence of incorrect initial sensor noise, five test cases have been defined in Tab. 4.3 where the initial values for sensor noise are too pessimistic in cases 1a/2a and too optimistic in cases 1b/2b.

4.3.3 Optimization Results

In the following, the results from the numerical optimization with the two different cost functions are presented.

[26] A choice $w = 0$ is doubtful since then the convexity of $J_{\mathrm{NS}}(\boldsymbol{z})$ is further impaired since the NIS contains less information about the system than the NEES. A weighted sum of NEES and NIS is in principle thinkable as long as $w \geq 0.1 \ldots 0.25$ holds approximately.

Results for the WEES- and WIS-based Cost Function

Starting with a simple initial choice of the four process noise and two sensor noise parameters, the numerical optimization provides the optimized parameters without further need for adjustment. The results are shown in Fig. 4.3 comparing the initial design with the optimized and final design. The improvement of the estimation quality is remarkable (cf. also Fig. 4.5, p. 95). At the same time also the average NEES $\bar{\varepsilon}_{\mathrm{NS}}$ is corrected from $2.10 \cdot 10^4$ to 1.79 and the SNIS $\bar{\nu}_{\mathrm{NS}}$ is reduced from 65.1 to 0.215 which means that the found estimator design is both informative and consistent.

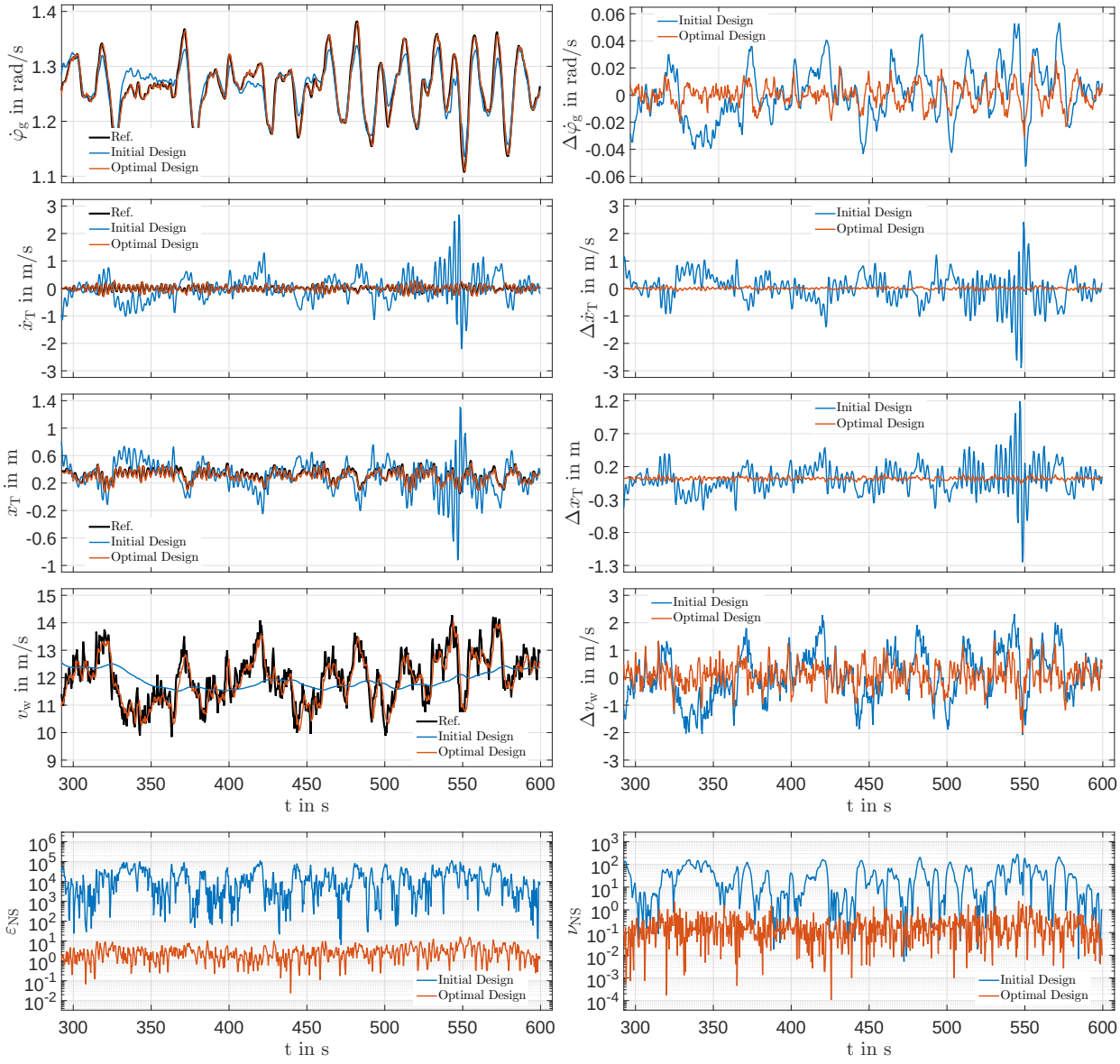

Figure 4.3: Optimization results for the test scenario shown in Fig. 4.2 using the cost function $J_{\mathrm{WS}}(\boldsymbol{z})$ and the predefined weights comparing the state estimates and additional performance variables for the initial design (light blue) against the optimal design (orange)

Results for the NEES- and NIS-based Cost Function

Next, the optimization results for the stochastic cost function are compared in Fig. 4.4 assuming a correct initial choice of the sensor noise. As mentioned before, only the process noise is optimized and the sensor noise is predefined. The estimation results and especially the reduced estimation errors highlight the improvement achieved with the numerical optimization. In addition, the average NEES $\bar{\varepsilon}_{\mathrm{NS}}$ is adjusted from the initial value $2.20 \cdot 10^4$ to 1.36 and the SNIS $\bar{\nu}_{\mathrm{NS}}$ is reduced from 25.8 to 0.184 which shows that the optimum is reached closely. Thus, the designed estimator works consistently with respect to the level of NEES and slightly pessimistically with respect to the level of NIS (cf. Fig. 4.4 at the bottom).

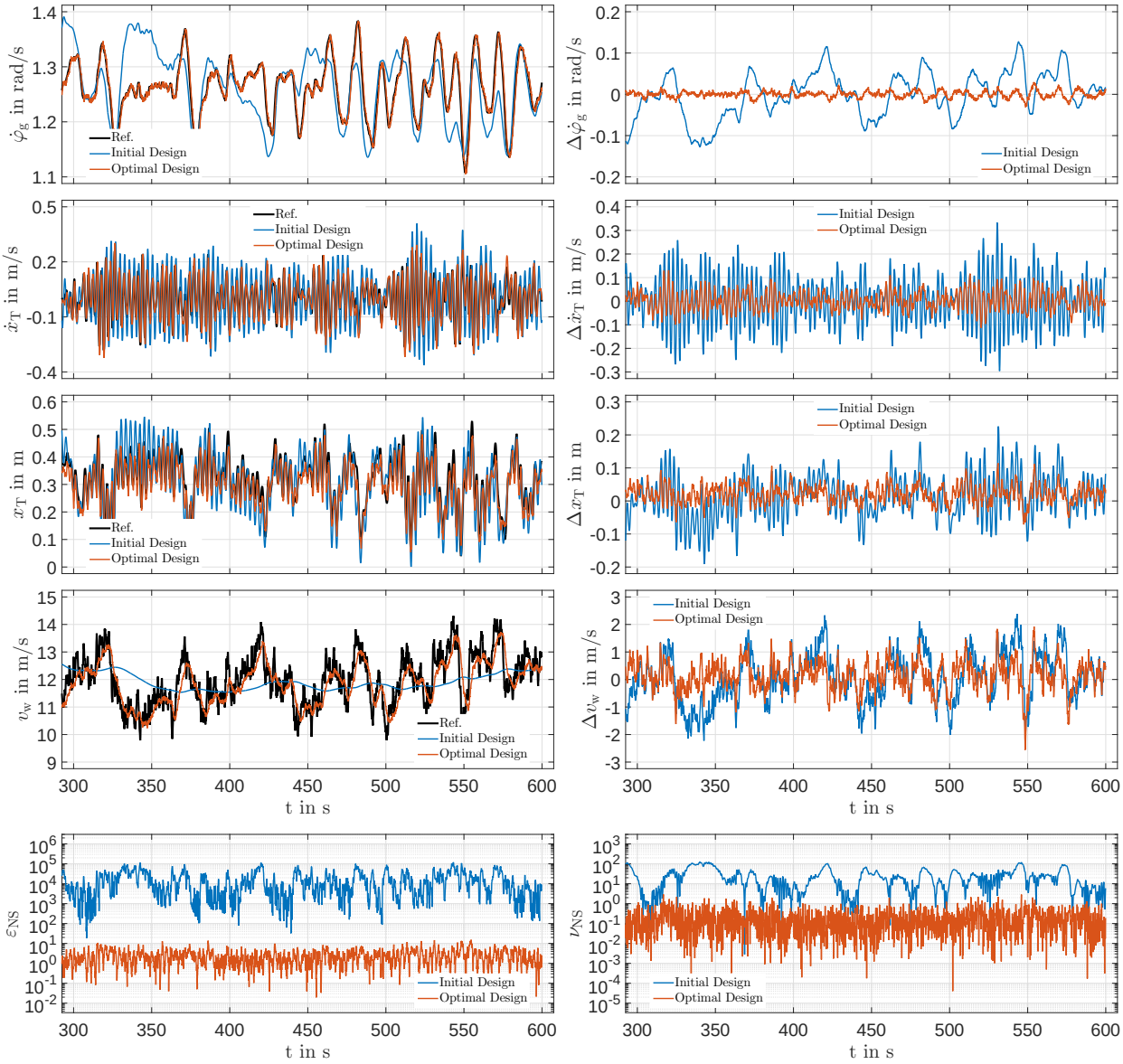

Figure 4.4: Optimization results for the test scenario shown in Fig. 4.2 using the cost function $J_{\mathrm{NS}}(\boldsymbol{z})$ and $w = 0.5$ with correct initial values for the measurement noises comparing the initial design (light blue) against the optimal design (orange) and the reference values (black)

Comparing the normalized estimation errors, Fig. 4.5 shows that the expected estimation accuracy is similar for both cost functions (although the performance of the initial designs obviously varies considerably).The normalized RMSE for the optimal design are in the same magnitude order. Additionally, the $\bar{\varepsilon}_{\mathrm{NS}}$ and the $\bar{\nu}_{\mathrm{NS}}$ are almost identical. Moreover, Tab. 4.4 summarizes the initial and optimal parameters for both cost functions (see p. 96).

It is interesting to note that a small estimation error and a consistent estimator are apparently connected since the same results are obtained for two fundamentally different cost functions. The big question remains whether this is a generic property of the estimator or only happens by accident. Unfortunately, this research question needs further investigation which is not in the scope of this thesis and thus cannot be answered at this point.

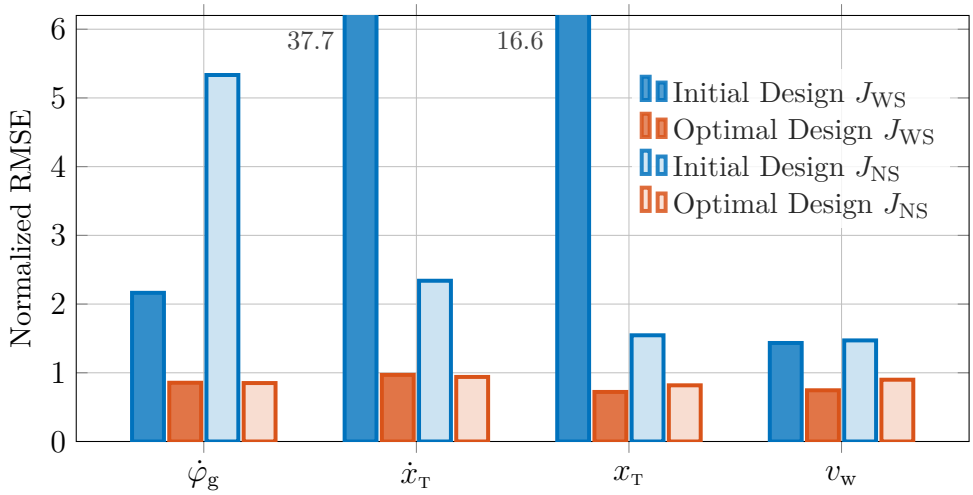

Figure 4.5: Normalized root mean-squared errors of the estimated state variables for the initial and optimal design for the two cost functions J_{WS} and J_{NS} based on the following normalization values: $\dot{\varphi}_{\mathrm{g}}^{\mathrm{n}} = 0.01\,\mathrm{s}^{-1}$, $\dot{x}_{\mathrm{T}}^{\mathrm{n}} = 0.04\,\mathrm{m/s}$, $x_{\mathrm{T}}^{\mathrm{n}} = 0.04\,\mathrm{m}$ and $v_{\mathrm{w}}^{\mathrm{n}} = 0.6\,\mathrm{m/s}$

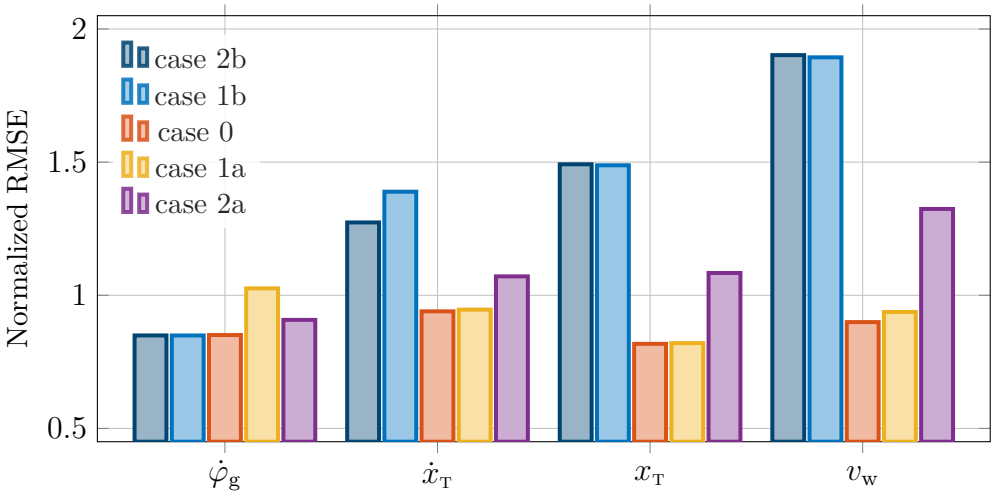

Figure 4.6: Normalized root mean-squared errors of the estimated states for different initial measurement noise variables (comparison for the optimal design) as provided in Tab. 4.2 using the following normalization values: $\dot{\varphi}_{\mathrm{g}}^{\mathrm{n}} = 0.01\,\mathrm{s}^{-1}$, $\dot{x}_{\mathrm{T}}^{\mathrm{n}} = 0.04\,\mathrm{m/s}$, $x_{\mathrm{T}}^{\mathrm{n}} = 0.04\,\mathrm{m}$ and $v_{\mathrm{w}}^{\mathrm{n}} = 0.6\,\mathrm{m/s}$

Table 4.4: Optimal and initial design parameters for the static filter design

noise variable physical unit	q_{11} in $(1/s)^2$	q_{22} in $(m/s)^2$	q_{33} in $(m)^2$	q_{44} in $(m/s)^2$	r_{11} in $(rpm)^2$	r_{22} in $(m/s^2)^2$
$J_{WS} \rightarrow$ init. design	10^{-6}	10^{-6}	10^{-6}	10^{-6}	10^{1}	10^{1}
$J_{WS} \rightarrow$ opt. design	4.48×10^{-4}	1.57×10^{-4}	4.95×10^{-3}	9.97×10^{-1}	1.24×10^{1}	10^{-3}
$J_{NS} \rightarrow$ init. design	10^{-6}	10^{-6}	10^{-6}	10^{-6}	1.24×10^{1}	10^{-3}
$J_{NS} \rightarrow$ opt. design	2.03×10^{-3}	1.30×10^{-3}	3.50×10^{-3}	7.67×10^{-1}	1.24×10^{1}	10^{-3}

Since the initial choice of sensor noise presumably has a great significance on the optimization result, this is investigated separately. The results for the five test cases are presented in Fig. 4.6 and Fig. 4.7. It can be seen that the best estimation results are obtained for the test case with the correct sensor noise. Admittedly, the effect is less dominant for the first state than the other ones. Interestingly, for the test case 1a (increased sensor noise by one magnitude order) the effect is less important but not for test case 1b (reduced sensor noise by one magnitude order). Furthermore, it is apparent to note that the incorrect sensor noise (which cannot be corrected by the optimization) prohibits that a consistent design with respect to the $\bar{\varepsilon}_{NS}$ and the $\bar{\nu}_{NS}$ can be reached (cf. Fig. 4.7 on the right). Tab. 4.5 provides the numerical values of the performance metrics for the two optimizations.

Table 4.5: Comparison of performance metrics for initial and optimal design

		case 2b	case 1b	case 0	case 1a	case 2a
initial design	$\bar{\varepsilon}_{NS}$	24318.067	22383.051	21999.875	21885.000	21469.211
optimal design	$\bar{\varepsilon}_{NS}$	55.509	7.315	1.358	0.670	0.292
initial design	$\bar{\nu}_{NS}$	138.068	65.280	25.764	11.122	3.796
optimal design	$\bar{\nu}_{NS}$	0.492	0.313	0.184	0.075	0.007
initial design	J_{NS}	19.329	19.407	19.712	19.119	17.660
optimal design	J_{NS}	2.908	1.673	0.849	0.764	2.062

Concluding, the illustrative example for static optimal design highlights the potential of an automated approach over an intuitive manual design procedure. It has been shown that the optimization reaches informative and consistent results independent of the cost function (if the sensor noise is known in advance). This is a remarkable result. If not known, the first cost function $J_{WS}(\boldsymbol{z})$ is recommended to be used although then the weighting matrices must be chosen explicitly.

4.4 Design of the Adaptive Filter

The adaptive design addresses in this context the ability of the filter to react to incorrect prior assumptions with regard to the noise covariance matrices. This may be due to false model or false filter parameters. The necessity for an adaptive component (within the observer architecture) is often indicated due to the following reasons:

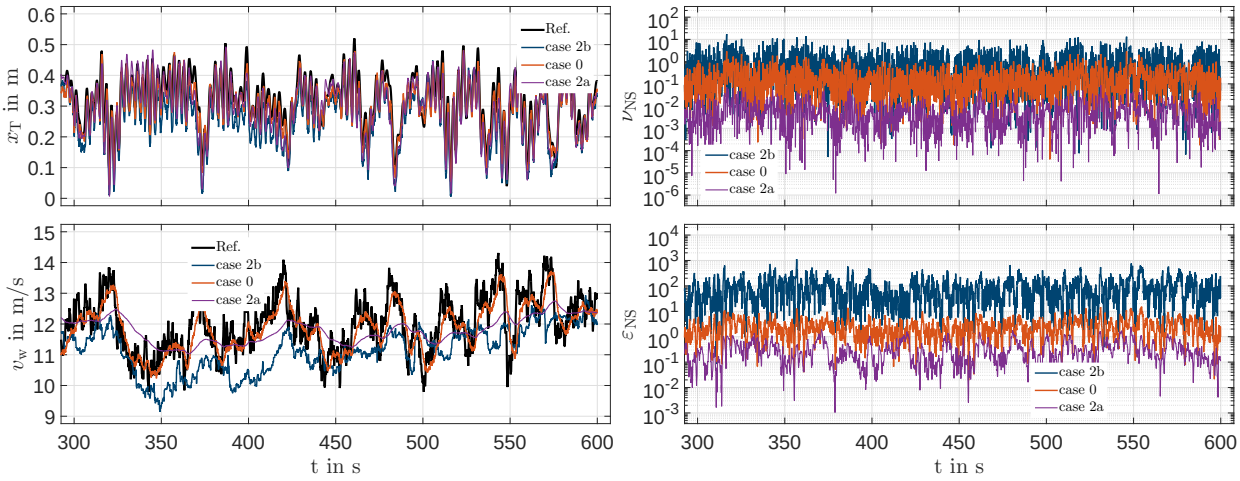

Figure 4.7: Illustrative optimization results for the test scenario shown in Fig. 4.2 using the cost function $J_{NS}(\boldsymbol{z})$ and $w = 0.5$ with different initial values for the measurement noises comparing the very optimistic (dark blue curve), the correct (orange curve) and the very pessimistic cases (purple curve) with the reference values (black)

- The initial choice of the noise covariance matrices differs from the actual values,
- the process or measurement noise covariances vary with respect to time,
- the noise covariances vary with respect to the current operating point.

As a disclaimer, the application of an adaptive component is recommended only if the noise environment is unknown or time-varying because the beneficial effect of noise adaptation might turn into the opposite if designed carelessly. The reason is that there is an additional filter feedback loop on the covariance level which may impair the estimation accuracy or even worse leads to filter divergence [2].

As a starting point for the following design approach, the optimization result from the classical filter (cf. Sect. 4.3.3) are reused.

4.4.1 Definition of the Dual Estimation Problem

The estimation problem basically stays the same as for the classical filter (cf. Sect. 4.3.1, p. 89 ff). Though, it is slightly modified in order to integrate the sensor noise state estimation into the adaptive filter design.[27] The advanced estimation problem considered hereafter denotes as follows:

> Estimate the nacelle fore-aft dynamics $\dot{x}_T$ and x_T as well as the unknown disturbance input v_w together with a monolithic CKF and estimate separately the

[27] Here only the sensor noise case is considered since the process noise is assumed to be correctly known after the first design step. If the process noise is uncertain, a similar slave filter can be used, cf. [166].

sensor noise covariances r_{11} and r_{22} with a second estimator (the slave filter)! Consider for this purpose an artificial time-varying sensor noise environment using realistic available data $\boldsymbol{u}_k$ and $\boldsymbol{y}_k$ (Fig. 4.8) as filter inputs and $\boldsymbol{x}_k$ for validation purposes from the FAST simulator. Moreover, use the optimal design parameters from Sect. 4.3.3 (p. 92 ff) as initial values for the sensor noise and as constant values for the process noise!

To fulfil this estimation task, the Model 3.4 is used again as simplified master filter model. The dynamic model of the slave filter is defined according to the selected parameters (cf. Sect. 2.5.2, p. 37 ff). The functionality of the automated design shall be demonstrated using an illustrative test scenario with an artificial sensor noise sequence (Fig. 4.8).

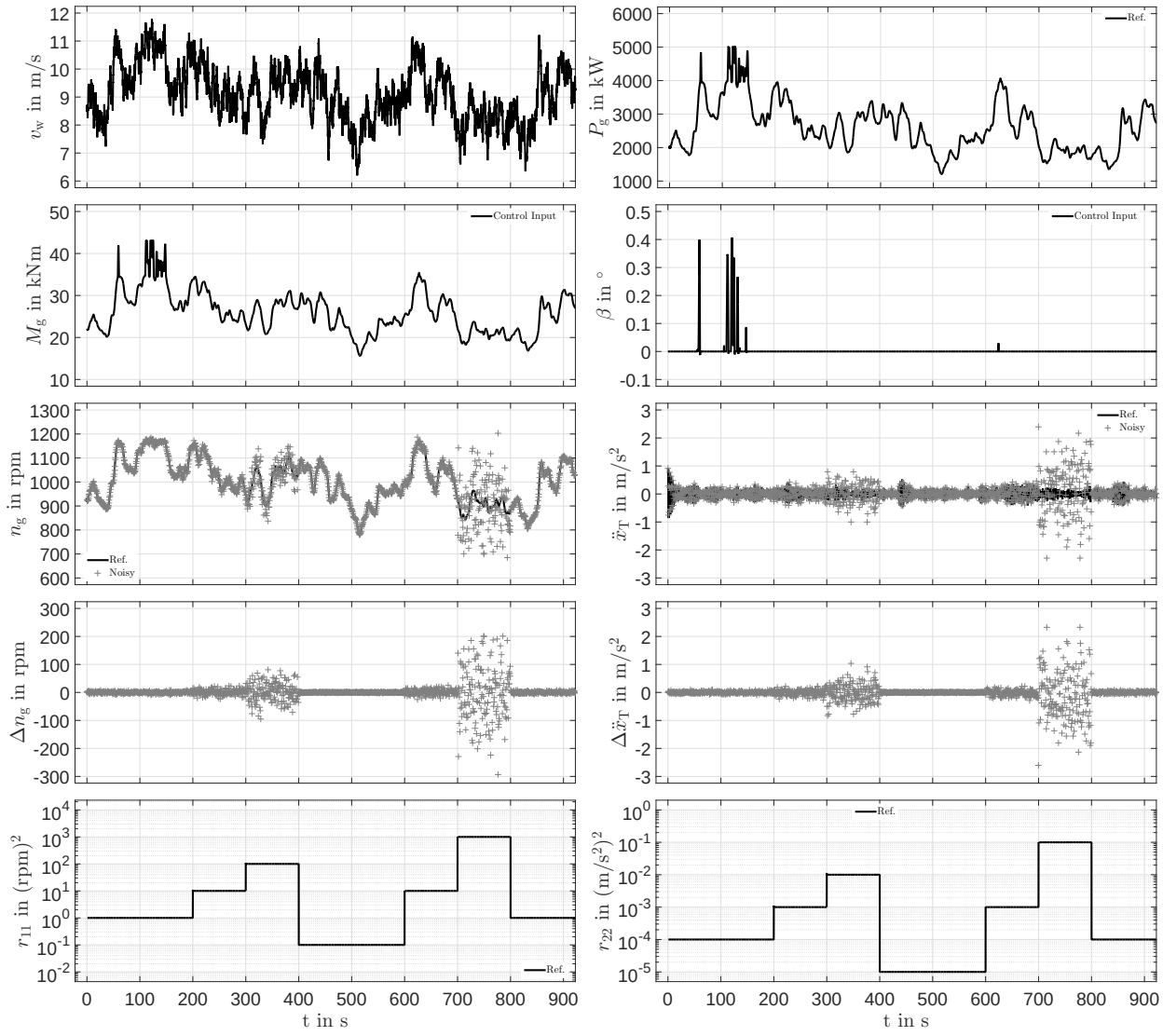

Figure 4.8: Test scenario is used to conduct the optimal design procedure for the adaptive component of the estimator. The average wind speed is 9 m/s and the sensor noises are chosen to be zero-mean with time-varying noise covariances, see Tab. 4.6

Therein, step changes in sensor noise covariances (as a worst case scenario) are applied to test the adaptive filter. In addition, the step changes are required to be able to optimize the slave filter's covariance matrices since otherwise no adaptation during operation is enforced. The selected variances alternate between different noise levels as defined in Tab. 4.6.

Table 4.6: Sensor noise scenario for adaptive filter design with step changes

$t_1 \leq t < t_2$	in s	$[0\,,200[$	$[200\,,300[$	$[300\,,400[$	$[400\,,600[$	$[600\,,700[$	$[700\,,800[$	$[800\,,900]$
$\mathrm{Var}\{n_\mathrm{g}\}$	in $(\mathrm{rpm})^2$	10^{+0}	10^{+1}	10^{+2}	10^{-1}	10^{+1}	10^{+3}	10^{+0}
$\mathrm{Var}\{\ddot{x}_\mathrm{T}\}$	in $(\mathrm{m/s^2})^2$	10^{-4}	10^{-3}	10^{-2}	10^{-5}	10^{-3}	10^{-1}	10^{-4}

4.4.2 Choice of the Decision Variables and Weights

The slave filter's noise covariance matrices $\mathbf{Q}_{\mathrm{s},k}$ and $\mathbf{R}_{\mathrm{s},k}$ represent the process and sensor noise of selected slave states which itself are the process or sensor noise of the master filter. Roughly speaking, these noise parameters represent the sensitivities of the master filter's process or measurement noise covariances. These quantities are even harder to guess than for the master filter. For the reasons earlier denoted, we propose to use the WEES- and WIS-based cost function $J_\mathrm{WS}(\boldsymbol{z})$ here because there is no reasonable approach to guess the variance of the fictitious sensor noise based on previous system knowledge which would be required for the NEES/NIS-based approach.

Since there are $n_x = 2$ noise states and $n_y = 2$ fictitious sensor outputs in the slave filter (which represent the diagonal elements of the empirical innovation covariance matrix), the total number of decision variables is $n_z = n_x + n_y = 4$. Hence, $\boldsymbol{z} \in \mathbb{R}^4$ must be designed prior to application of the slave filter. The initial decision variables denote

$$\boldsymbol{z}^0_\mathrm{WS} = \left[q^0_{\mathrm{s},11}\ q^0_{\mathrm{s},22}\ r^0_{\mathrm{s},11}\ r^0_{\mathrm{s},22}\right]^\mathrm{T} \tag{4.26}$$

and are set to

$$q^0_{\mathrm{s},11} = 10^{-8}\,,\ q^0_{\mathrm{s},22} = 10^{-8}\,,\ r^0_{\mathrm{s},11} = 10^{-4} \text{ and } r^0_{\mathrm{s},22} = 10^{-4}\,. \tag{4.27}$$

Different compared to the master filter design, the cost function $J_\mathrm{WS}(\boldsymbol{z})$ in Eq. (4.12), p. 84, is not based on the system states $\boldsymbol{x}_k$ and outputs $\boldsymbol{y}_k$ but rather uses the diagonal elements of the process noise covariances $\boldsymbol{q}_k$ and the sensor noise covariances $\boldsymbol{r}_k$. The reason is that the solver gets stuck in local minima or does not have a minimum at all (depending on the scenario) when using $\boldsymbol{x}_k$ and $\boldsymbol{y}_k$. The change to the process and sensor noise states improves substantially on the lack of convexity.

Since there is no information about the true process noise available whatsoever, it cannot be considered as ground truth information in this optimization problem. For that reason,

the focus rests on the known artificial sensor noise only (cf. Fig. 4.8 bottom). The weighting matrices are chosen as follows

$$\mathbf{W}_x = \operatorname{diag}\left\{\begin{bmatrix}0 & 0\end{bmatrix}^{\mathrm{T}}\right\} \quad , \quad \mathbf{W}_y = \operatorname{diag}\left\{\begin{bmatrix}10^{-4} & 1\end{bmatrix}^{\mathrm{T}}\right\} \tag{4.28}$$

$$\mathbf{W}_{\dot{x}} = \operatorname{diag}\left\{\begin{bmatrix}0 & 0\end{bmatrix}^{\mathrm{T}}\right\} \quad , \quad \mathbf{W}_{\dot{y}} = \operatorname{diag}\left\{\begin{bmatrix}10^{-4} & 1\end{bmatrix}^{\mathrm{T}}\right\} \tag{4.29}$$

and therefore only the trajectory of $\boldsymbol{r}_k$ is weighted in the cost function.

4.4.3 Optimization Results

The optimization problem is solved again using the interior-point solver. The results are collected in Fig. 4.9. The left-hand side shows the complete scenario while the right-hand side focuses upon the extreme noise level only, see time window $t \in [680, 820]\,\mathrm{s}$.

In these plots, the classical CKF is included also as a reference for the low noise environment (dark blue curves). The initial design (light blue) and the optimal design (orange curves) depict the performance of the master-slave filters with different noise parameters. There are several things that are remarkable at a glance:

- First, the optimal MS-CKF estimates the mean value of the different sensor noise levels (cf. Tab. 4.6) very accurately. This is especially true for the higher noise levels as the curves for the estimates $\hat{r}_{11}$ and $\hat{r}_{22}$ reveal (see $t \in [200, 400]\,\mathrm{s}$ and $t \in [600, 800]\,\mathrm{s}$).

- The initial MS-CKF design is obviously insufficient to track changes in sensor noise fast enough. Though, one could argue that such instant changes are not very likely in practice. Since this behaviour is tunable, the filter design can meet these turbine specific requirements.

- For the low noise environment in the regions $t \in [0, 200]\,\mathrm{s}$ and $t > 800\,\mathrm{s}$ smaller deviations from the true noise level are observed. These become even larger at (effectively) zero noise level ($t \in [400, 600]\,\mathrm{s}$) and the slave filter even diverges for the generator speed noise r_{11}. Hence, it is practically important to limit the noise estimates to the physical lower bound (as done here as well).

- The poor performance of the optimal MS-CKF, when there is barely noise at all, is directly visible in the NEES $\varepsilon_{\mathrm{NS},k}$ for $t \in [400, 600]\,\mathrm{s}$. However, this does not affect the estimation accuracy at all but just the consistency of the estimator.

- The severe effect of larger sensor noise on the state estimates is noted unanimously for the states x_{T} and v_{w} in the time series plots (see $t \in [300, 400]\,\mathrm{s}$ and $t \in [700, 800]\,\mathrm{s}$). In addition, the Fig. 4.10 provides the normalized RMSE which highlights the improvement of estimation accuracy when a slave filter component is integrated.

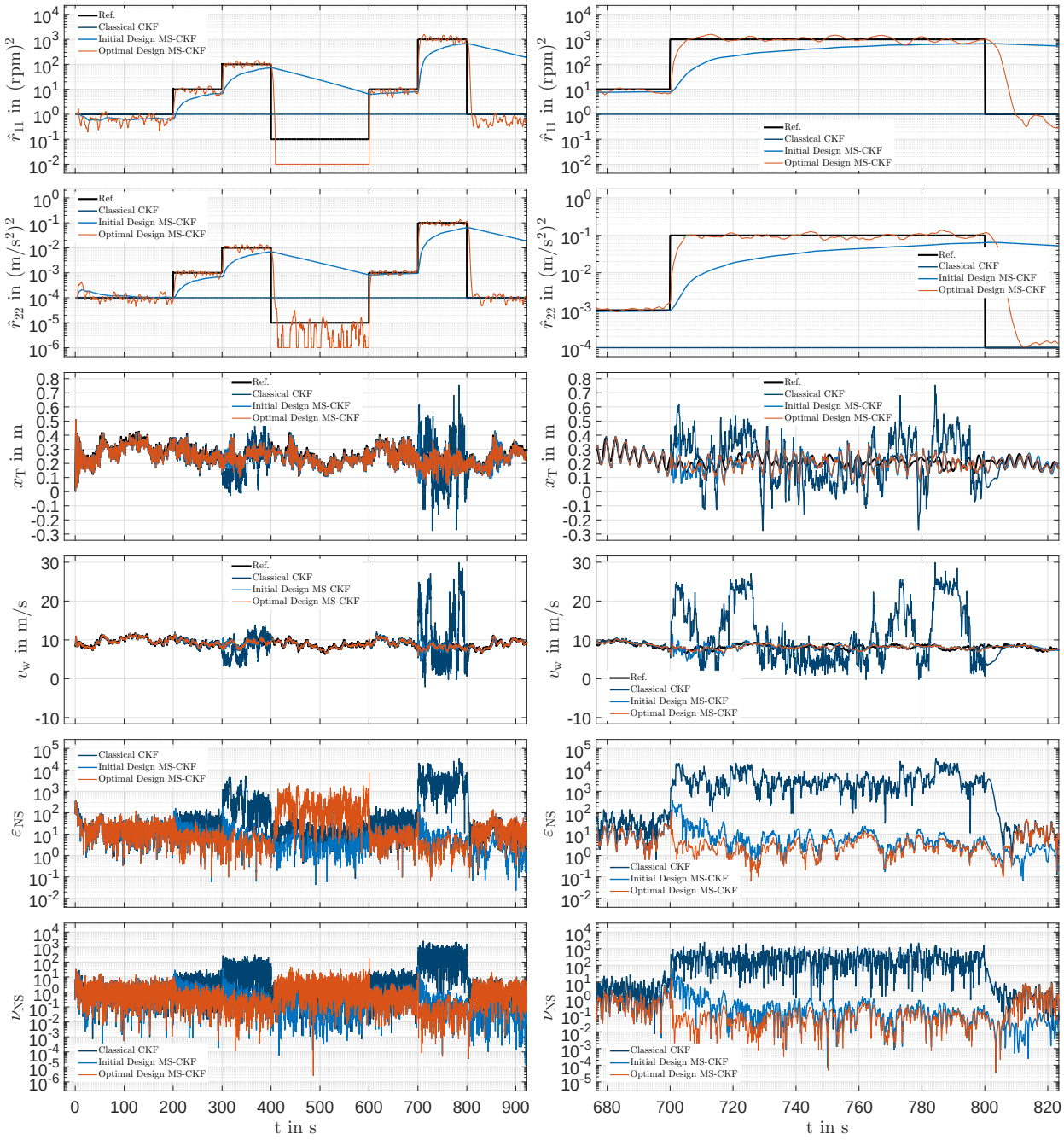

Figure 4.9: Optimization results for the test scenario shown in Fig. 4.8 using the cost function $J_\mathrm{WS}(\boldsymbol{z})$ and the predefined weights comparing the state estimates, noise estimates and further KPI for the classical filter (dark blue) and the adaptive filter's initial design (light blue) against the optimal design (orange)

- The slave filter has also a beneficial effect on the filter consistency which is observed for both NEES and NIS in Fig. 4.9 (right-hand side). Additionally, the MS-CKF appears to detect the critical noise environment since both KPI (NEES and NIS) decrease with increased noise. This means anything less but that the estimator becomes more pessimistic about its own estimates (which is the correct reaction to increased noise).

- And finally, this before-mentioned effect is perfectly true if one looks at the state estimates x_T and v_w. Both become recognizably worse (compared to before and after the step change) yet still more accurate than the CKF estimates at the same time.

In a nutshell, the MS-CKF shows a very robust performance over a large range of sensor noise levels and therefore appears to be a good choice if the noise environment varies strongly over time. More simulation results for an acid test scenario with time-varying noise are presented in Sect. 5.4.3 (p. 118).

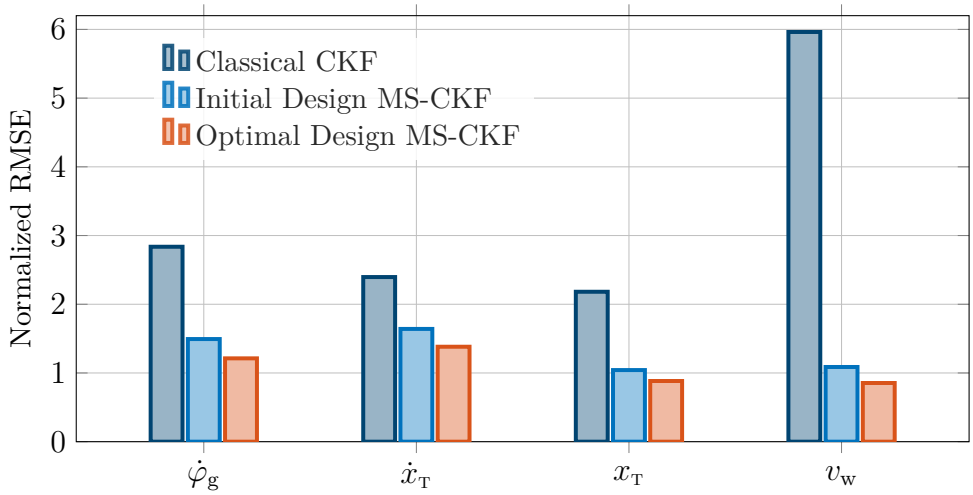

Figure 4.10: Normalized root mean-squared errors of the estimated states for the classical filter as well as the initial and optimal design of the novel MS-CKF based on the following normalization values: $\dot{\varphi}_\mathrm{g}^\mathrm{n} = 0.01\,\mathrm{s}^{-1}$, $\dot{x}_\mathrm{T}^\mathrm{n} = 0.04\,\mathrm{m/s}$, $x_\mathrm{T}^\mathrm{n} = 0.04\,\mathrm{m}$ and $v_\mathrm{w}^\mathrm{n} = 0.6\,\mathrm{m/s}$

4.5 Summary and Conclusions

This chapter summarizes the proposed novel filter design methodology based on numerical optimization that pursues the goal to facilitate a highly automated filter design. This is desired in order to reduce the expenditure for designing SPKF components repeatedly and also to improve the expectable estimation quality at the same time with minimum effort.

This automated design is based on the numerical solution of a well-defined optimization problem. For this purpose, the generic optimization problem was formulated first which was then customized step by step to the specific estimation problem. The procedure was exerted exemplarily for the classical as well as the consecutive adaptive filter design.

Two alternative cost functions were introduced which integrate the estimation error and the innovation differently into the cost function. The first $J_\mathrm{WS}(\boldsymbol{z})$ uses predefined weighting matrices for $\boldsymbol{e}_k$ and $\boldsymbol{v}_k$ which gives the control designer a lot of freedom to influence the optimization to his or her favour. The second cost function $J_\mathrm{NS}(\boldsymbol{z})$ uses the filter-computed covariances for the estimation error and the innovation as weights. Therewith, the need for

selecting design weights is needless and no weights must be selected manually at all. This is on the one hand desirable however on the other hand it was found that this elimination of predefined and fixed weights reduces the number of estimable noise variables inevitably. Thus, only process or measurement noise can be optimized while the respective other one must be known in advance to achieve a filter design that is consistent and with minimum estimation error.

The design approach was successfully applied and demonstrated for a representative wind turbine state estimation problem. The transfer to different problems is possible though has not been the focus of the present chapter.

Concluding, both cost functions have their benefits and draw-backs. Yet, they share the advantage that the commonly used trial-and-error method is fully or partially replaced by an optimization algorithm. It was shown that this approach works quite well. With the presented novel methodology the question of how to choose the filter parameters is circumvented. Hence, it enables the direct usage for automatic filter generation and parameter design in a software tool chain.

5 Simulation Studies and Performance Assessment

The evaluation of the Kalman filter's performance under preferably realistic conditions is an important step prior to field testing. Extensive simulation studies shall investigate every critical aspect that may deteriorate the filter's performance in the final practical application. Relevant aspects for wind turbines are the effects of changing operating conditions and model inaccuracies as well as the sensor noise environment.

5.1 Introduction

This chapter presents the results of the wind turbine simulation study. This numerical study is useful to assess the filter performance prior to application in the field. That is powerful because within simulation different effects can be analysed in a secluded fashion and, in addition, the ground truth (real values) is known and can be used for performance assessment.

In order to conduct such a detailed study, a tailored simulation environment has been developed which is called the filter performance assessment tool, or FPA-tool for short. It is designed to investigate all relevant aspects including different wind scenarios, various sensor errors, several filter algorithms, filter designs, different estimation problems and the optimization of filter parameters (among many others). The FPA-tool with its main modules is introduced in Sect. 5.2. Afterwards, Sect. 5.3 (p. 109 ff) presents the test scenarios considered in this simulation study. Since there is a variety of possible scenarios, the focus lays on a few important effects such as average wind speed, turbulence intensity, initial filter design, sensor noise and all of them combined (called acid tests). These are discussed for the estimation problems defined in Chap. 4 (p. 81 ff). Hereafter, Sect. 5.4 (p. 113 ff) discusses the estimation results of the different filters and filter designs for the previously defined scenarios. In particular, the differences in filter performance between the classical and the adaptive SPKF are highlighted and discussed.

Since the choice of the examinable aspects in the field of wind turbines is very broad, further simulation results for advanced estimation problems are stressed in Sect. 5.5 (p. 123 ff). These results have already been introduced in the author's previous publications and are put here together into the larger context. Finally, the last Sect. 5.6 (p. 133 ff) gathers the relevant outcomes of this chapter and discusses their overall meaning.

5.2 Filter Performance Assessment Tool

The filter performance assessment tool (in short FPA-tool) is the newly developed simulation environment to evaluate the state estimators holistically. It is programmed entirely in Matlab/Simulink and has initially been designed for offline filter testing only. It is intended to extend this engineering tool also for online assessment in the future.

The FPA-tool consists of several modules to address the needed functionality and also the desired flexibility to deal with the various test cases in a comfortable manner. These modules include the following:

- The test scenario generation module (FPA/G module),
- the filter parameter optimization module (FPA/O module),
- the filter execution module (FPA/E module),
- the performance evaluation module (FPA/P module), and
- the visualization module (FPA/V module).

In this context, Fig. 5.1 provides a schematic overview of the FPA-tool's architecture. After every successful completion of a module, the new results are appended to the scenario's archive. In that way, also intermediate results are backed up to check the generated (filter) results for correctness, plausibility and performance.

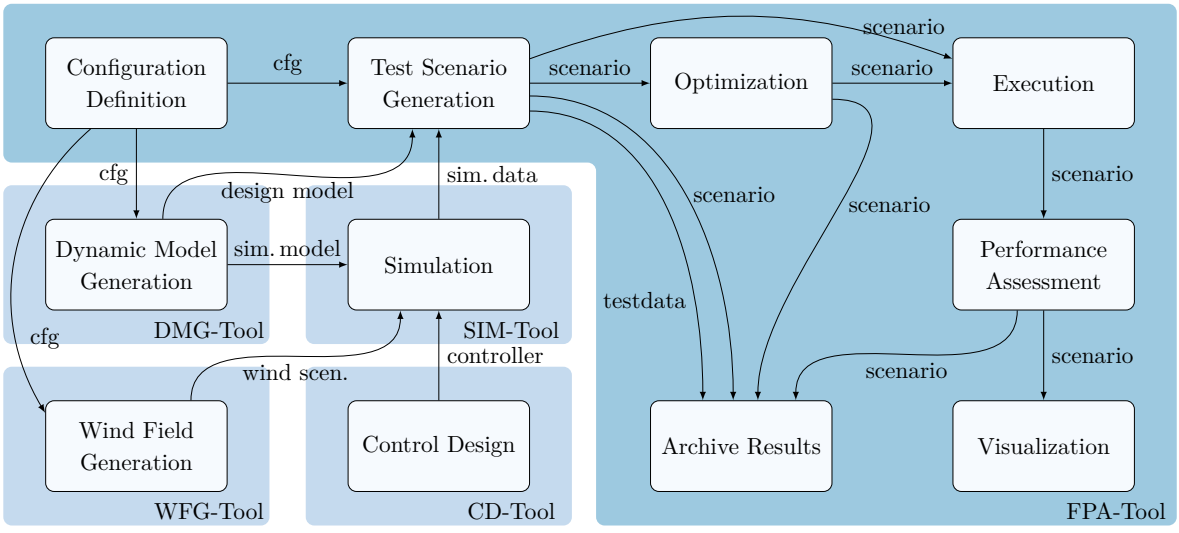

Figure 5.1: Structural sketch of the developed FPA architecture and its connections to other developed tools for simulation/design model generation, for wind field generation, for controller preparation and for wind turbine simulation

5.2.1 Scenario Generation Module

This module is capable to generate (almost) arbitrary test scenarios from a single configuration file. In order to do so, the FPA/G module encompasses the following adjustable properties to define the scenarios:

- The properties of the wind scenarios
- The choice of wind turbine model
- The filter algorithm and its parameters
- The definition of the estimation problem
- Different sensor noise environments
- The optimization settings for static and adaptive design

The properties of the wind scenario include (but are not limited to) the average wind speed, the turbulence intensity and the wind shear. For that purpose, the various wind scenarios have been generated with NREL's TurbSim [94] using different random seeds. These scenarios are made available (accessible) from a large local data base.

The choice of wind turbine model (for generation of test data) implies that different design and simulation models are available from a model library. On the one hand, these representations are design models (cf. Sect. 3.4) with different granularities and modelling details (ranging from one to four states up to 28 states including blade flap and edge dynamics). Therewith, the models are used to generate the specific test data (control inputs, control outputs, states and performance variables) for filter testing in the first place. On the other hand, high-fidelity simulators like NREL's FAST8 [96] are well suited as realistic simulation models.[28] In addition, it is beneficial that these simulators have been approved for wind turbine certification by renowned certification bodies.

The choice of the filter algorithm and its parameters include the various types of SPKF, different adaptation rules and their parametrization. The implemented SPKF types encompass the CKF, the CDKF and the UKF as well as their square-root versions (SR-CKF, SR-CDKF and SR-UKF). Moreover, the implemented adaptation rules/algorithms are the master-slave (MS) filter and the various realizations of the MLE including (but are not limited to) the versions of Maybeck, Mohamed & Schwarz, Vepa and Sun. The filter parameters include the initial states, the initial error covariance, the initial process and measurement noise covariance of both standard and adaptive filter components (see also Sect. 5.2.2 and Chap. 4). Obviously, these filter parameters depend on the settings of the estimation problem, discussed hereafter.

[28] FAST8 can be used in the FPA tool and also a variety of different design models are available.

The definition of the specific estimation problem (to be investigated) is necessary to allow for the selection of states to be estimated, selection of measurements available, selection of model parameters to be estimated. An extensive model library is required which covers a variety of process models with different number of states and output models with the sensor configurations. The latter models are an integral part of the FPA tool.

Finally, the different noise scenarios are easily incorporated using scenario configuration files where sensor noise is parameterizable. These sensor noise scenarios selectively include random noise (Gaussian white noise), outliers, biases and random walks which are all possibly chosen to be time-dependent (cf. Sect. 5.3.3). The noise scenario are also applicable and considered for the filter parameter optimization in Chap. 4.

5.2.2 Filter Parameter Optimization Module

The FPA/O module is mainly used to perform the automatic filter design (cf. Sect. 4.3 and 4.4). If activated, this module must also be parametrized by the scenario configuration file. The following settings are applicable:

- Choice of cost function and free weighting factors
- Choice of optimization algorithm
- Choice of the time window(s) for optimization
- Choice of the process and measurement noise diagonal elements (to be optimized)
- Choice of lower and upper bound for the above covariance (optional)
- Choice of weights for the states and outputs as well as their respective first time derivatives
- Choice of weighting factor w to balance NIS and NEES
- Further options related to optimization algorithm (tolerances, function evaluations, etc.)

The FPA-tool is capable of generating various scenarios and combinations from the above settings to assess the effect on the optimization results in detail. This shall guarantee that the estimator works under any adverse conditions.

5.2.3 Filter Execution Module

This module performs the filter simulation for the previously generated scenarios. These original scenarios (generated by the FPA/G module) may be modified if the FPA/O module has been executed before. In this case, both designs (initial and optimized) are available

in the archived results. The FPA/S module is also used as an integral part in the optimization module (cf. Sect. 5.2.2) since the system simulation is required to assess the objective function numerically.

5.2.4 Performance Evaluation Module

After all scenarios have been completed successfully, the estimation results are evaluated based on the performance criteria in Sect. 2.4 (pp. 27 ff.). These criteria include (but are not limited to) the state estimation errors, the innovations, the RMSE, the NIS and the NEES, as well as the estimation error covariances. The results of the performance evaluation are appended to the scenario's archive for the sake of completeness and potential later use.

5.2.5 Visualization Module

The FPA/V module prepares the simulation data and generates afterwards automatically the requested plots for documentation purposes. Moreover, it displays a selection of all relevant variables for each scenario in order to

- compare and check the generated test scenarios for correctness,
- compare the initial design versus the optimal filter design,
- compare and assess the estimation results w.r.t. the selected scenario properties.

The module provides all requested plots as graphics in various file formats to facilitate the barrier-free post-processing. Most of the graphics in Chap. 4 and 5 have been generated with the FPA/V module.

5.3 Definition of the Test Scenarios

The test scenarios are defined in order to study different effects on the filter performance (occurring in practice). These effects may impair the expected estimation results and accuracy directly. To understand their individual relevance, they are investigated in this simulation study first in a preferably secluded manner. Thereafter, the effects are combined to investigate worst case scenarios and interdependencies. The scenarios are distinguished in the following four kinds:

1. Scenarios related to the wind field, meaning the effects of operating range and wind conditions on filter performance.
2. Scenarios related to the filter algorithm, the filter design and the internal model.

3. Scenarios related to the sensor data, meaning incorrect and/or deficient sensor information (which is the only information/input data for the filter.

4. Scenarios considering all of the above simultaneously (denoted as acid tests).

In the following, the different dimensions (scenarios) to compare classical and adaptive filters are introduced.

5.3.1 Average Wind Speed

The average wind speed (AWS) is considered in order to investigate the actual operating point of the turbine. The designated testing points

$$v_{\mathrm{m}} \in \Big[6, 9, 12, 15, 18\Big] \,\mathrm{m/s} \tag{5.1}$$

are chosen in order to cover the partial load regime (PLR, R2), the transition region (TR, R2.5) as well as the full load regime (FLR, R3). The vertical shear exponent is set to $\kappa = 0.2$. The sensor noise environment is considered to be low (cf. Tab. 4.1, p. 91) while outliers and faults are omitted. The turbulence intensity is set to a medium level with TI = 0.1 (cf. Sect. 5.3.2). Fig. 5.2 shows the wind scenarios for the five different AWS. Each AWS scenario has been selected arbitrarily from a pool of ten scenarios and stands representative for the respective average wind speed. The results for the other nine wind fields with the same statistical properties (AWS and TI) are omitted here for sake of brevity. The above wind scenarios are tested with the two filters from Chap. 4 each for two filter designs. The first filter is designed for low sensor noise levels and the second filter for zero sensor noise environment.

5.3.2 Turbulence Intensity

Besides the AWS, the second statistical property of the wind scenarios is the turbulence intensity (TI). The definition writes

$$\mathrm{TI} \equiv \frac{\mathrm{std}(v_{\mathrm{w}})}{\mathrm{mean}(v_{\mathrm{w}})} \tag{5.2}$$

which is the ratio of standard deviation divided by the average wind speed v_{m}. The more TI increases, the more the scenario covers also global effects due to rapid changes between operating regimes and operating points. Consequently, the nonlinearity of the system has a greater impact on the filter performance which shall be tested explicitly. The following set of TI is chosen:

$$\mathrm{TI} \in \Big[0.05, 0.1, 0.2\Big] \,. \tag{5.3}$$

These are denoted as low, medium and high TI levels hereafter. The average wind speed is set to $v_{\mathrm{m}} = 12\,\mathrm{m/s}$ so that all wind speeds $v_{\mathrm{w}}(t)$ remain in the operating range for power

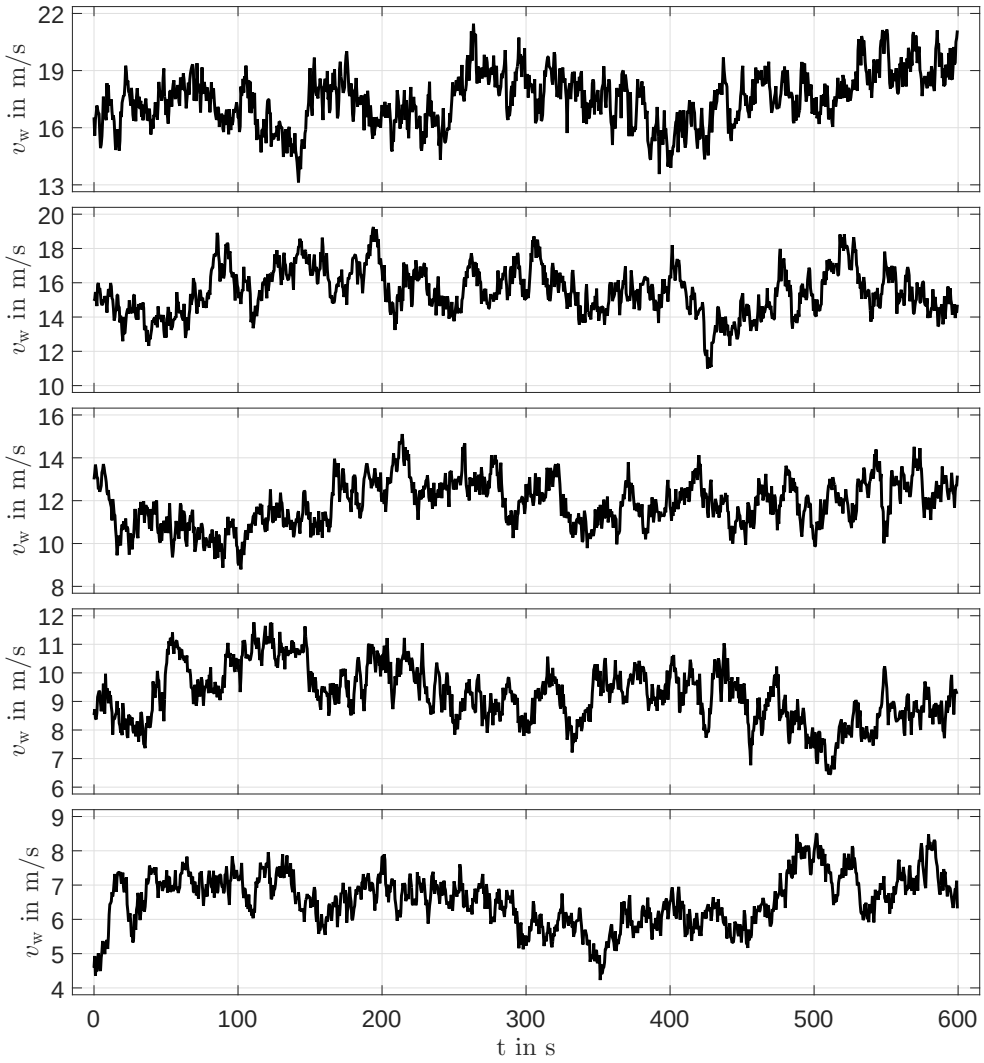

Figure 5.2: Test scenarios for five different average wind speeds with the same TI = 0.1 (10%)

production. Furthermore, multiple change between the PLR and FLR are included, hence both control actions are present in each scenario. Again, the sensor noise is considered as low while outliers and faults are omitted in this test scenario. Fig. 5.3 shows the wind scenarios for the three different turbulence intensities. As can been seen, the stochastic wind field is similar for all three scenarios. That is why, one original wind field has been rescaled to meet different TI levels. That is, the effect of change in TI is secluded from other effects like the random changes in wind speed (which is intended here). Moreover, for the first scenario with low TI all wind speeds stay within a range of $\pm 2\,\mathrm{m/s}$ (thus more or less local). Other than that, the other two scenarios cover with $12 \pm 4\,\mathrm{m/s}$ and even $12 \pm 8\,\mathrm{m/s}$ the whole operating range within a single 10-minute wind scenario.

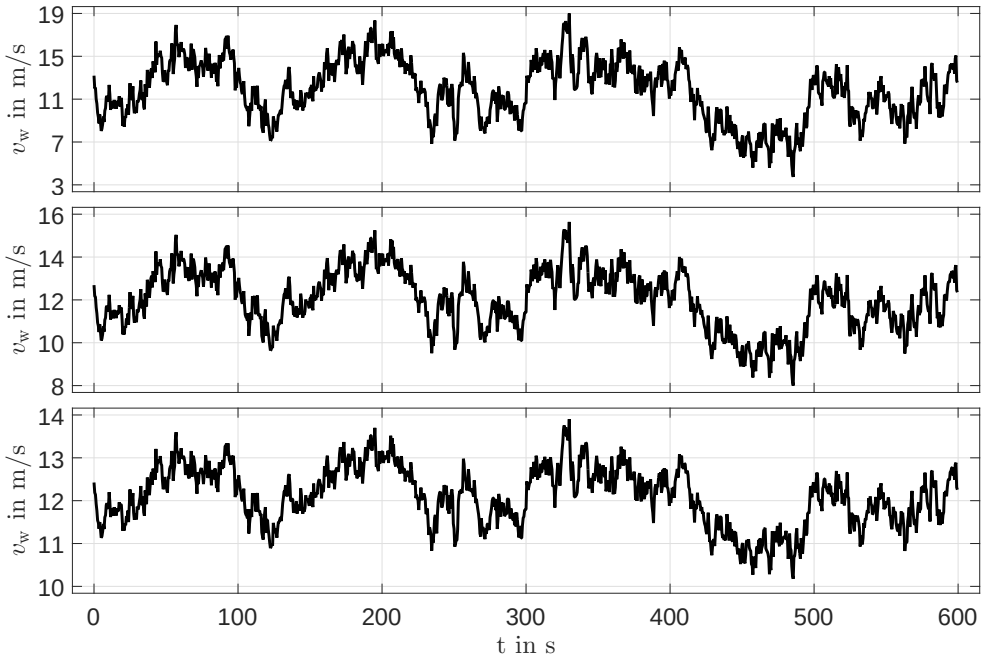

Figure 5.3: Test scenarios for three different turbulence intensities from low TI (bottom) to high TI (top) with $v_m = 12\,\mathrm{m/s}$

5.3.3 Acid Test Scenario

The final scenario is used to assess the full capability of the designed adaptive estimators to cope with multiple practical effects. In that way, the superiority over the standard filter shall be probed.

In this scenario, step changes in the Gaussian noise variances are considered as a critical case. Fig. 5.4 shows the plots for the relevant quantities for an average wind speed $v_m = 12\,\mathrm{m/s}$ with TI = 0.1. The variances alternate between the different levels listed in Tab. 4.1 (p. 91) and the tested noise sequence is described in Tab. 5.1 (p. 112) with the specific time intervals. As can be seen from the figure, the aerodynamic rotor power P_a alternates strongly around a mean value close to 5.000 kW. The control actions with M_g and β enforces that the power production does not exceed the nominal generator power P_g in average. Moreover, the disturbed sensor information is shown at the bottom. Similar scenarios for $v_m = 9\,\mathrm{m/s}$ and $v_m = 15\,\mathrm{m/s}$ are provided in App. B, cf. Fig. B.6 (p. 152).

Table 5.1: Sensor noise scenario with step changes in variances

$t_1 \leq t < t_2$	in s	$[0\,,200[$	$[200\,,350[$	$[350\,,500[$	$[500\,,650[$	$[650\,,900]$
$\mathrm{Var}\{n_g\}$	in $(\mathrm{rpm})^2$	10^{+1}	10^{+2}	10^{+1}	10^{+3}	10^{+1}
$t_1 \leq t < t_2$	in s	$[0\,,250[$	$[250\,,400[$	$[400\,,550[$	$[550\,,700[$	$[700\,,900]$
$\mathrm{Var}\{\ddot{x}_T\}$	in $(\mathrm{m/s^2})^2$	10^{-3}	10^{-1}	10^{-3}	10^{-2}	10^{-3}

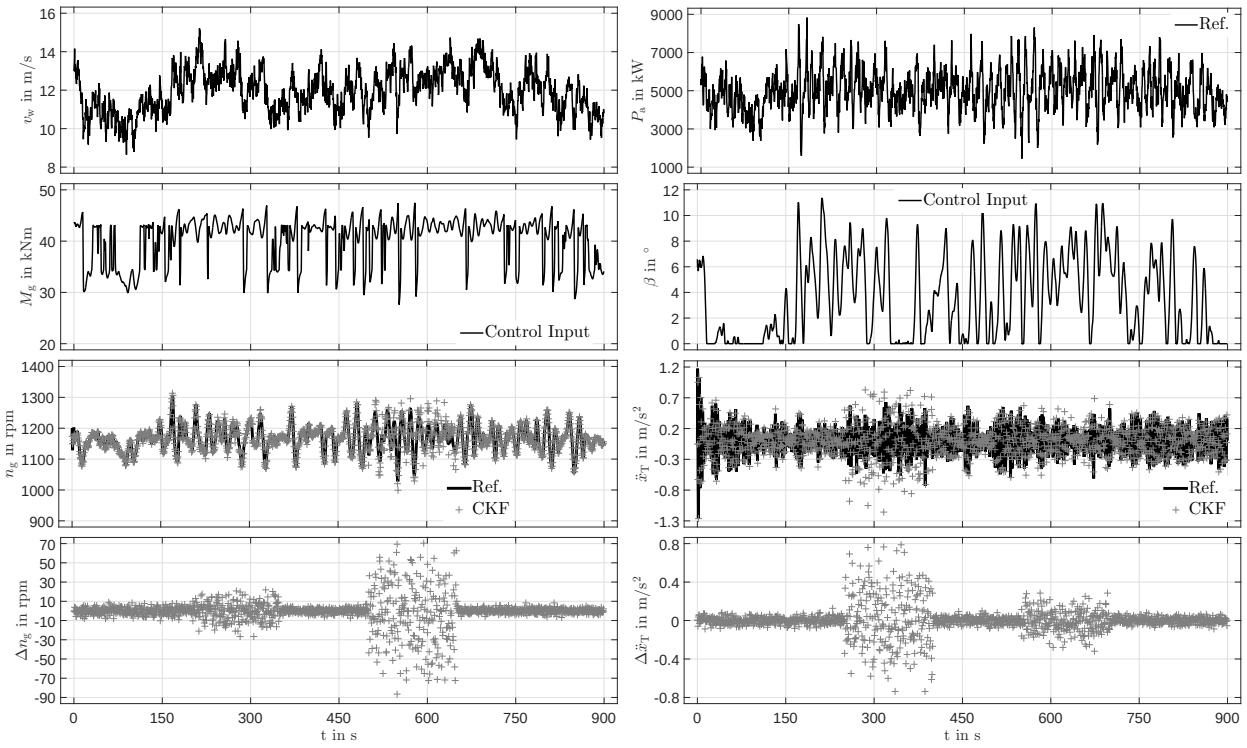

Figure 5.4: Acid test scenario for $v_m = 12\,\mathrm{m/s}$ with $\mathrm{TI} = 0.1$ with time-varying sensor noise on n_g and $\ddot{x}_T$. The noise levels are provided in Tab. 5.1

5.4 Filter Performance Analysis

This section focuses on the performance analysis for the filters designed in Chapter 4. For this purpose, the test scenarios defined in Section 5.3 are applied to probe the different effects separately. Finally, both filter types (the classical and adaptive filter) are put to the acid tests.

5.4.1 Average Wind Speed

Due to the system's nonlinearity, the operating point of the wind turbine influences significantly the system dynamics. Moreover, the control strategy changes from partial to full load regime (et v.v.). In the transition region or with higher turbulence intensity, these changes occur more frequently.

Hence, the influence of average wind speed (AWS) need to be investigated in greater detail. In particular, two critical states are compared for the five AWS and the two different designs of measurement noise covariance matrices are tested as mentioned earlier.

First, Fig. 5.5 on p. 114 compares the wind speed estimates with the reference value for both estimators and filter designs (Fig. B.1 shows the corresponding estimation errors). From these plots, the following observations are made:

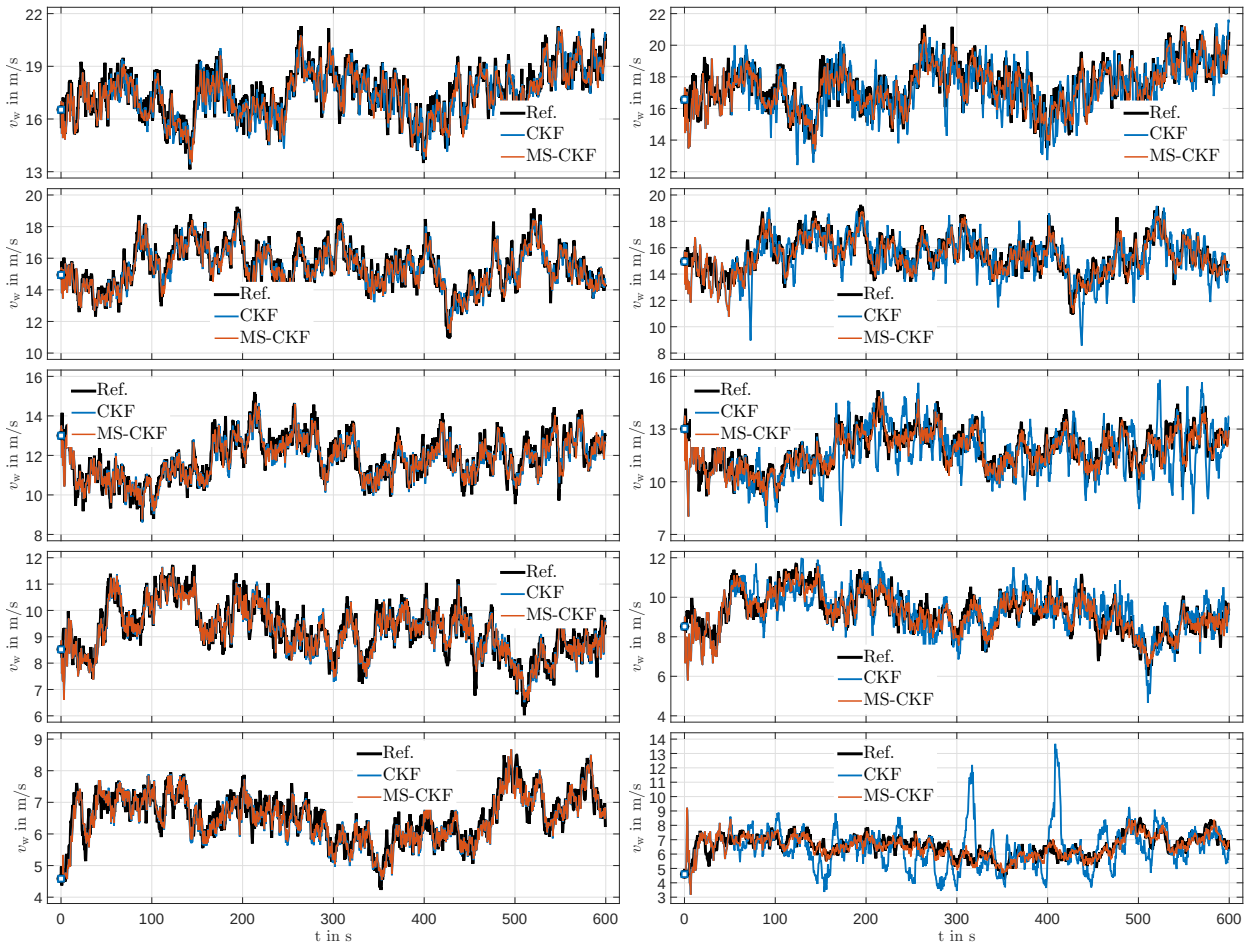

Figure 5.5: Estimation results from the AWS scenario $v_m \in [6, 9, 12, 15, 18]$ m/s (from the bottom up) with the low sensor noise environment comparing the wind speed estimates (left: designed for low noise level, right: designed for zero noise), cf. Fig. B.1

- If the initial design has been done with correct noise assumptions (left-hand side of Fig. 5.5), both filters provide similar estimation results that are almost independent of the chosen AWS. Though with increasing AWS, both estimates of CKF and MS-CKF slightly deviate from another.

- If the initial design does not match the correct sensor noise level (right-hand side of Fig. 5.5), the performance of the standard filter degrades seriously while the adaptive filter tackles the altered noise environment by adapting the master filter's parameters (cf. Fig. 5.7).

- Before the adaptation of the MS-CKF starts at $t = 50$ s, both filters show exactly the same results (as expected). Yet for $t > 50$ s, this changes and the adaptive SPKF with parallel slave filter shows always the same or better performance than the standard CKF. That is especially relevant if noise assumptions are incorrect (or the noise is time-varying, cf. Sect. 5.4.3).

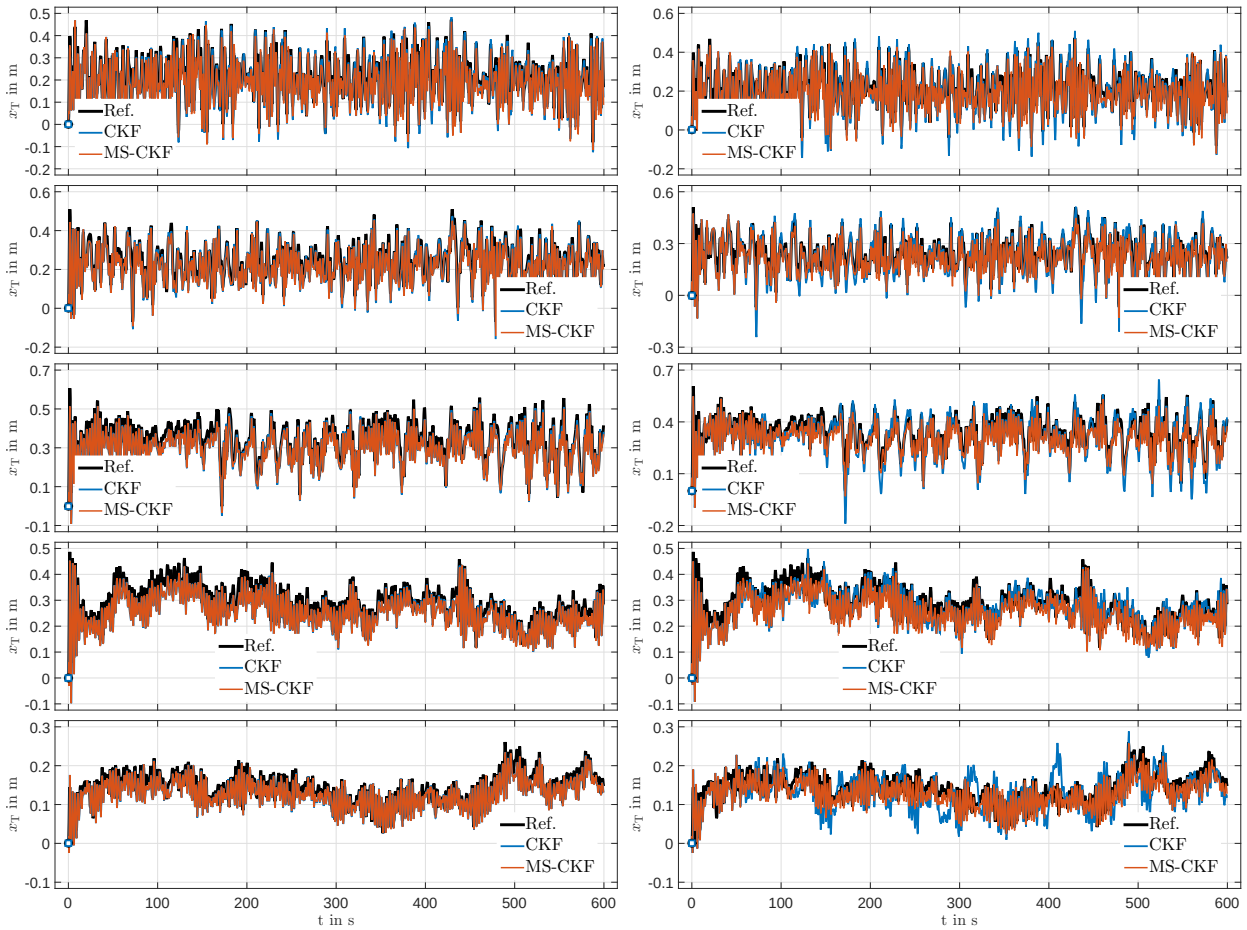

Figure 5.6: Estimation results from the AWS scenario $v_m \in [6, 9, 12, 15, 18]$ m/s (from the bottom up) with the low sensor noise environment comparing the nacelle position estimates (left: designed for low noise level, right: designed for zero noise).

- A relatively small error in measurement covariance of only one magnitude order leads to a critical decline in estimation accuracy.

Looking at the second state estimate (nacelle position) in Fig. 5.6, similar observations as for the wind speed estimate can be drawn from the plots. Though, the sensitivity to an error in noise covariance is apparently less critical for the nacelle position estimate and is more likely to be influenced by the degradation of the wind speed estimate (consult Fig. B.2, p. 150, for a direct comparison of the estimation errors).

The reason for the improved filter performance under uncertain model assumptions, compared to the CKF, is explained by the adaptive filter component. That is why, the MS-CKF estimates online its optimal measurement noise covariances as displayed in Fig. 5.7. The left-hand side shows the noise estimates for correct initial design and the right-hand side for the erroneous design. The constant values of the CKF are added as reference. The following things are remarkable:

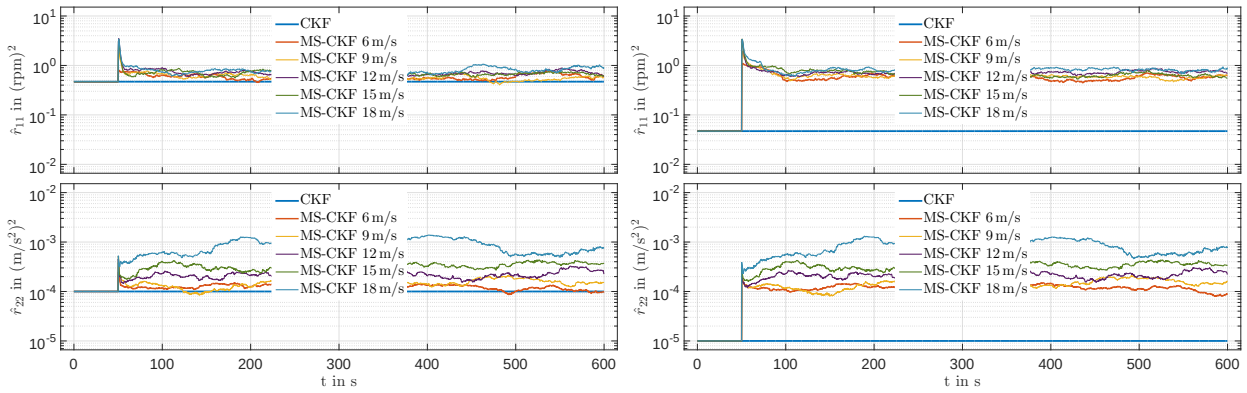

Figure 5.7: Estimation results from the AWS scenario with low sensor noise (left: designed for low noise level, right: designed for zero noise) comparing the measurement noise states, cf. Fig. B.3

- The MS-CKF adapts very quickly to the new noise environment. The noise estimates for both filter designs show almost the same curves for $t > 50\,\mathrm{s}$ (but they are not identical). Hence, these estimates appear to be independent of the actual initial design (cf. Figs. 5.7 and B.3, p. 150).
- The first noise estimate $\hat{r}_{11}$ (related to the generator speed sensor) varies only within a small range for different AWS. Thus, there is no need for adaptation related to a changing operating point.
- The second noise estimate $\hat{r}_{22}$ (related to the nacelle acceleration sensor) varies in a range of one magnitude order. Yet, this does not effect the performance very much as can be seen in Fig. 5.6 (left-hand side).

In conclusion, it is clear that the MS-CKF is superior to the standard CKF if and when the assumed noise level is initially incorrect. While the CKF is stuck with its static design parameters, it cannot tackle different noise environments. Conversely, the MS-CKF adapts quickly in less than 10 s (which depends on the window length N used to compute the empirical covariance matrices, cf. Sect. 2.4.2). Therewith it outperforms the standard filter.

5.4.2 Turbulence Intensity

The turbulence intensity (TI) is the second aspect to be considered in the simulation study (cf. Sect. 5.3.2). With increasing TI, the wind speed changes more rapidly and hence the operating regime is also swapped more frequently.

In this section, the research comprises and focuses on the following questions:

- Does the TI level impair the filter's performance?
- What effect does a poor initial design have at different TI levels?

- Does an adaptive filter concept help here to improve performance?

For this purpose a turbulent windfield with $v_\mathrm{m} = 12\,\mathrm{m/s}$ with different TI is taken into consideration (cf. Fig. 5.3). The obtained simulation results for these scenarios are shown in Fig. 5.8 and Fig. 5.9. Therein, the plots are arranged from low TI to high TI (from the bottom up). First, Fig. 5.8 shows the wind speed estimates. It is observed that the TI has only little effect if the filter design is correct (left-hand side). Then, there are only small deviations between the CKF and the MS-CKF, though increasing with TI level. Looking at the second state estimate (nacelle position) in Fig. 5.9, the deviations are in general notably small(er) and dependent only slightly on the filter design and algorithm.

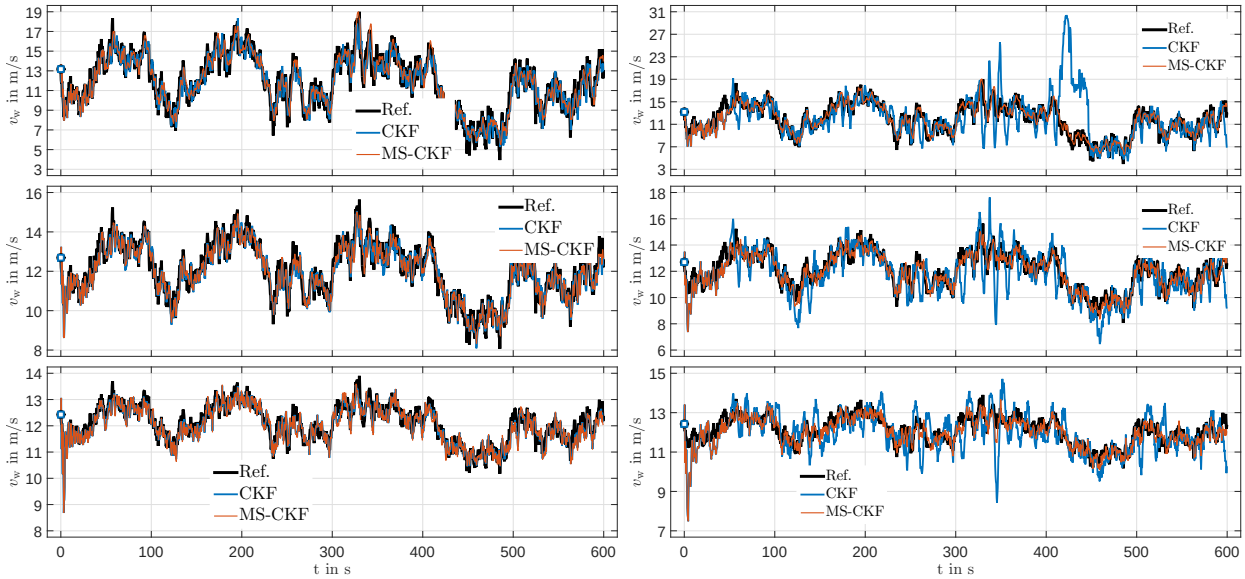

Figure 5.8: Estimation results from the TI scenario with low sensor noise comparing wind speed estimates (left: designed for low noise level, right: designed for zero noise), (cf. Fig. B.4, p. 151)

For sake of completeness, Fig. 5.10 presents the measurement noise estimates of the different MS-CKF performances. The constant noise of the CKF is included at a time as light blue curves. The master-slave filter shows again its superiority to cope with poor design since the noise estimates converge to similar mean values. Differences between changing TI levels are observed, though these do not interfere with the estimation accuracy very much.

Considering again the above research questions, it has been found that the TI level does not influence the filter performance very much. For this reason, an adaptation due to changing TI is not required. By contrast, the correct initial design is a far more important leverage. Finally, the simulation results suggest that the increasing TI is partly interpreted by the adaptive filter component as increasing sensor noise (which the filter seeks to compensate).

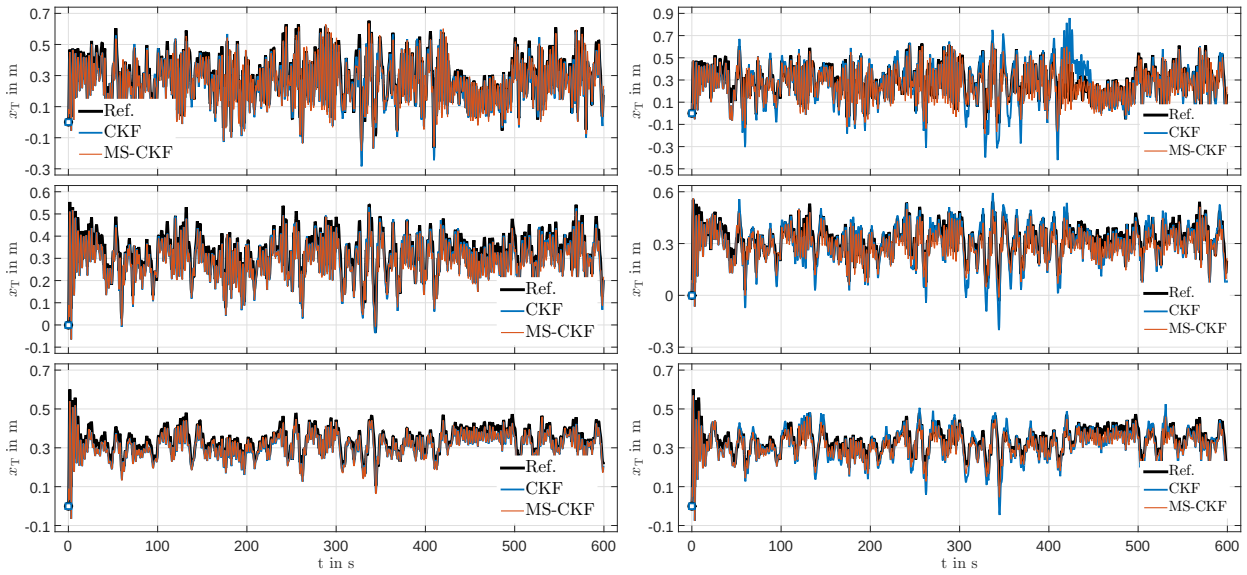

Figure 5.9: Estimation results from the TI scenario with low sensor noise comparing nacelle position estimates (left: designed for low noise level, right: designed for zero noise), (cf. Fig. B.5, p. 151)

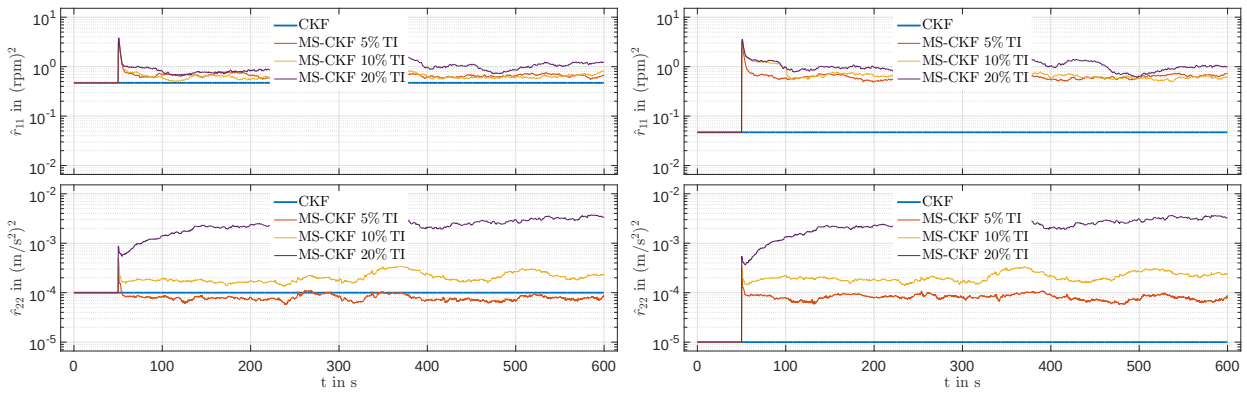

Figure 5.10: Estimation results from the AWS scenario for a low sensor noise environment with the same scaling of axis for enhanced comparability (left: designed for low noise level, right: designed for zero noise), cf. Fig. B.3

5.4.3 Acid Test Scenarios

In the following, the results for the acid test scenario, defined in Sect. 5.3.3, are presented. First, the state estimates are compared and thereafter, the performance variables are discussed. Fig. 5.11 shows the estimation results for the complete simulation time interval $t \in [0\,,\,900]\,\mathrm{s}$ and Fig. 5.12 contains the details for the interval $t \in [450\,,\,600]\,\mathrm{s}$. In both figures, the four relevant states are compared for classical and adaptive filters. The following general statements can be drawn from these plots:

- The results for the classical filter (CKF) show that the estimation error depends strongly on the right choice of the filter noise covariances.

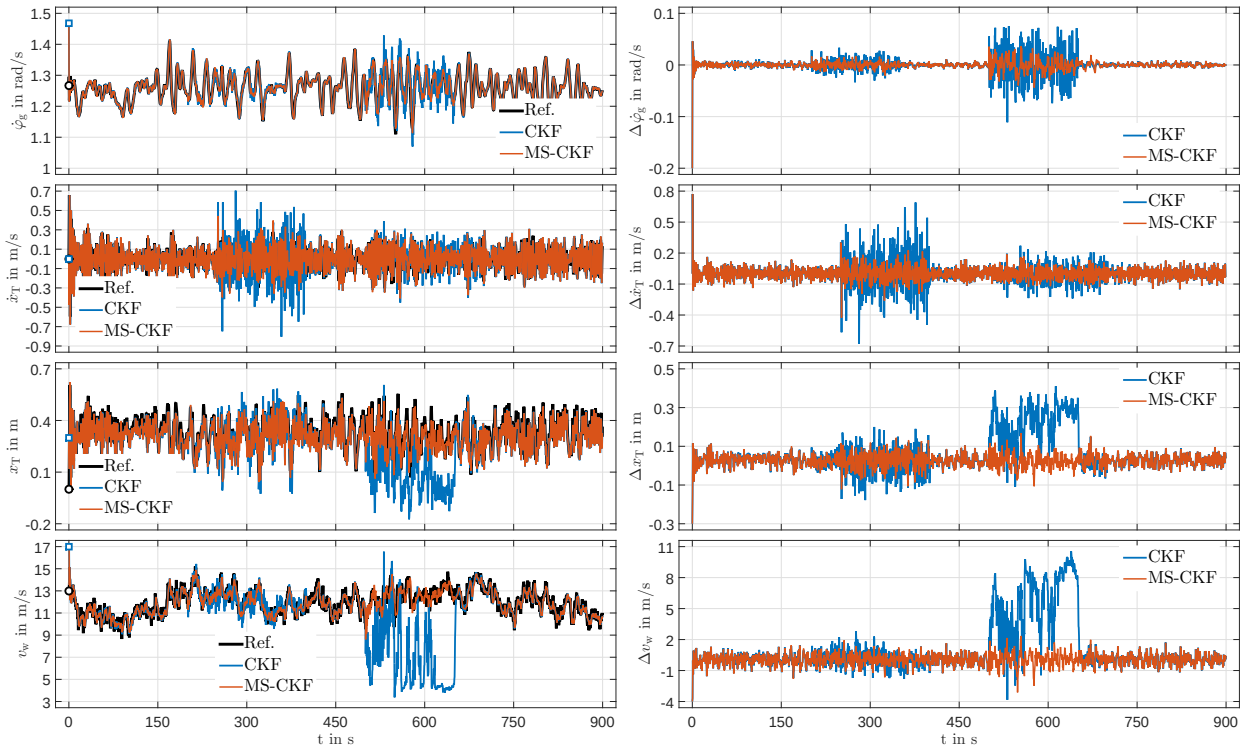

Figure 5.11: Estimation results from the acid test scenario with $v_m = 12\,\mathrm{m/s}$ and TI $= 0.1$ as well as time-varying sensor noise for the time window $t \in [0\,,\,900]\,\mathrm{s}$ (left: state estimates, right: estimation error), cf. Fig. B.7 on p. 153

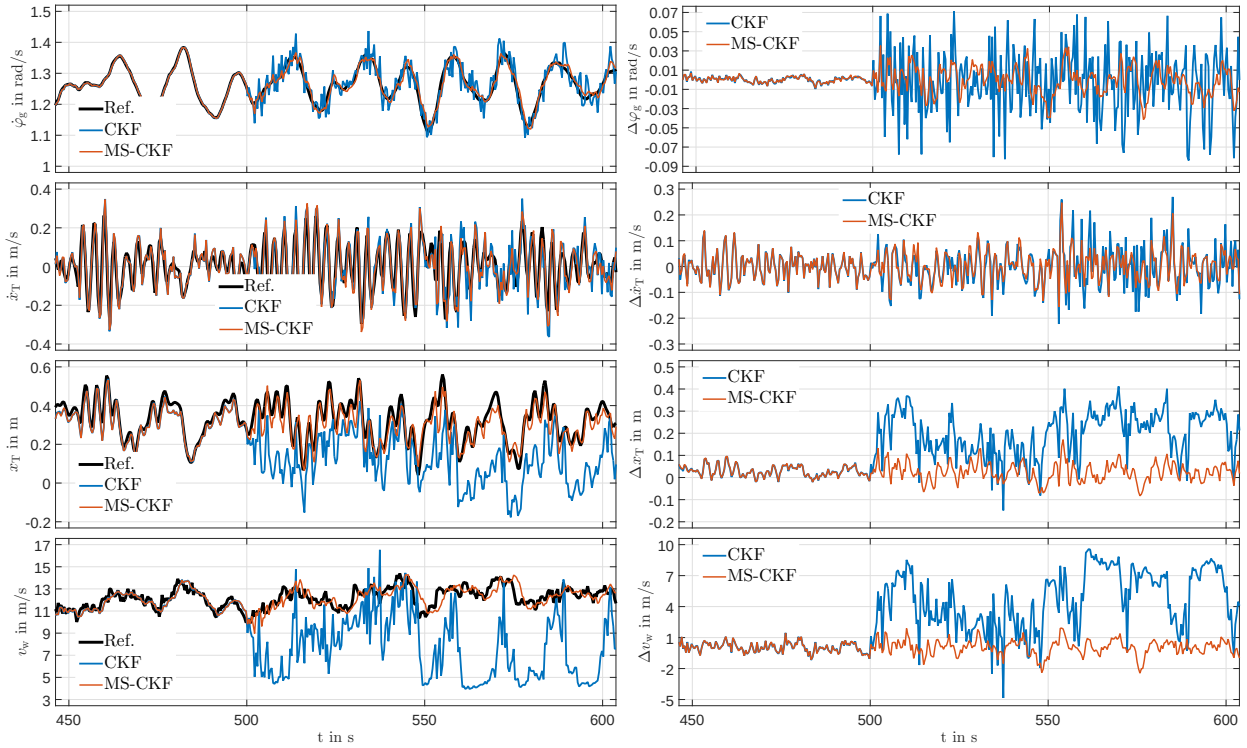

Figure 5.12: Estimation results from the acid test scenario with $v_m = 12\,\mathrm{m/s}$ and TI $= 0.1$ as well as time-varying sensor noise for the time window $t \in [450\,,\,600]\,\mathrm{s}$ (left: state estimates, right: estimation error), cf. Fig. B.8 on p. 153

- The estimation error (CKF) changes to the worse (degrades) whenever measurement noise jumps to different (increased) level.
- For certain estimates, the CKF tends to diverge temporarily though it is able to recover when returning to the previous/initial low noise environment. This is due to the constrained handling feature, implemented for both filters, cf. [171] for details.
- By contrast, the estimation errors for the MS-CKF remain almost unaffected by the step changes in sensor noise. Filter divergence does not arise in any scenario (cf. Figs. B.7 to B.10, p. 153 f).
- The estimation performance of CKF and MS-CKF is almost identical in the low noise environment ($t < 200\,\mathrm{s}$ and $t > 650\,\mathrm{s}$). Thus, the adaptive filter remains *silent* if no noise adaptation is needed.

Since there is an obvious improvement of filter performance gained from the adaptive filter component, the next Fig. 5.13 focuses on the noise estimates and filter performance variables. On the left-hand side, the diagonal elements of the measurement noise covariance matrix $\mathbf{R}_k$ are shown as a function of time. Regarding the CKF, there is (evidently) no estimated value thus the static design is shown for comparison. On the right-hand side, the NEES and NIS values are shown evolving over time.

Since both diagonal elements are released in the MS-CKF to be estimated, they ought to be adapted in an automated fashion which apparently happens. The following statements can be made for the results presented in Fig. 5.13:

- The MS-CKF adapts quickly to the new noise environment. It takes only 50 s to reach the new steady-state which corresponds to choice of the receding time window used in the slave filter.
- After a transition period, the estimated covariances remain at a virtually constant value until the next step occurs.

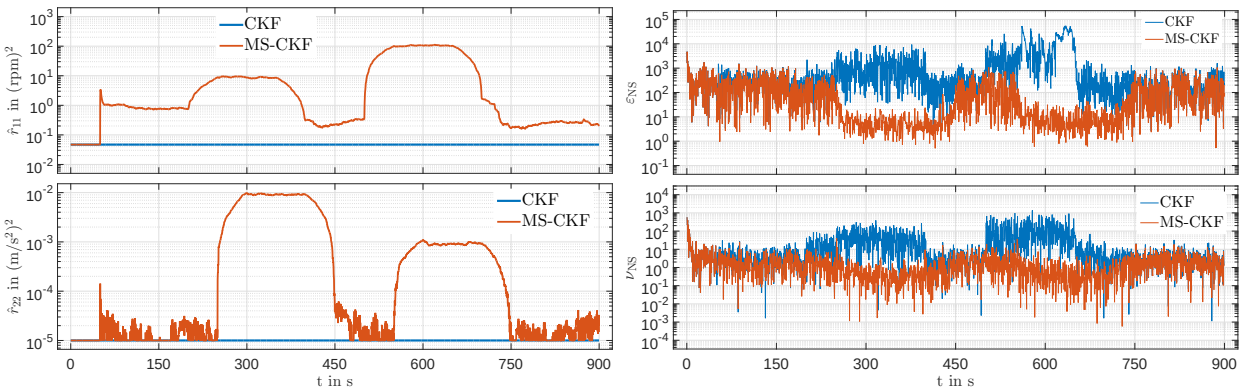

Figure 5.13: Estimation results for the acid test scenario with $v_{\mathrm{m}} = 12\,\mathrm{m/s}$ and TI = 0.1 as well as time-varying sensor noise (left: noise estimates, right: performance variables)

- The slave filter reacts sensitively to the actual noise level which can be seen in the different heights of the hat-shaped curves in Fig. 5.13 (cf. also Fig. 5.4). No filter divergence of the MS-CKF is observed (despite the increased risk due to the innovation covariance feedback).

- The noise covariances are estimated along with the dynamic states which indicates (not proves) that all quantities are observable at the same time using the standard sensor equipment (a practical proof could be achieved by a joint state and filter parameter observability analysis using for instance empirical observability Gramian matrices (EOG).[29]

- Hence, the additional information contained in the innovation sequence $\boldsymbol{v}_k$ is not only useful to correct the state estimates, but also to adapt the sensor noise estimates based on the empirical innovation covariance $\hat{\mathbf{P}}_k^{yy}$, cf. Eq. (2.27). This can be seen especially in the NIS $\nu_{\mathrm{NS},k}$ (Fig. 5.13) which remains almost unaffected by the time-varying noise.

Besides the estimates for the dynamic states and outputs, the filter also produces several additional (but dependent) estimates as by-product. That is, the employed internal wind turbine model evaluates intrinsically further variables such as mechanical turbine loads, aerodynamic coefficients and model parameters. These depend directly on the estimated dynamic states since this filter can only provide in total n_x independent values/variables.

The Figs. 5.14 and 5.15 give an overview of selected performance variables of interest (for instance used for monitoring purposes). Therein, the estimates for aerodynamic rotor thrust F_{T}, rotor torque M_{a} and rotor power P_{a} as well as generator power P_{g} are displayed on the left-hand side. Moreover, the generator acceleration $\ddot{\varphi}_{\mathrm{g}}$, the tip-speed-ratio λ, the power coefficient C_{P} and trust coefficient C_{T} are shown on the right-hand side. Generally speaking, most of these quantities are not directly measurable and thus, knowing them from state estimation (using virtual sensors, sometimes referred to as softsensors) provides an added value for the turbine operator.

The following is observed from the graphic illustrations in Figs. 5.14 and 5.15:

- Likewise (as for the state estimates) the adaptive filter provides estimates which are far more robust under the influence of disturbances.

- The estimation accuracy is in general very good, given the circumstances that this test assumes adverse conditions. The wind speed estimate has a significant effect on all other variables.

- The estimation quality is reduced with increased noise which is expectable. Yet, the performance loss is far smaller for the MS-CKF than for the standard CKF. Hence,

[29] Cf. [195, 49, 78, 53, 77] for an introduction to EOG and refer to [177, 169] for the application of EOG to observability analysis for wind turbines and wind farms.

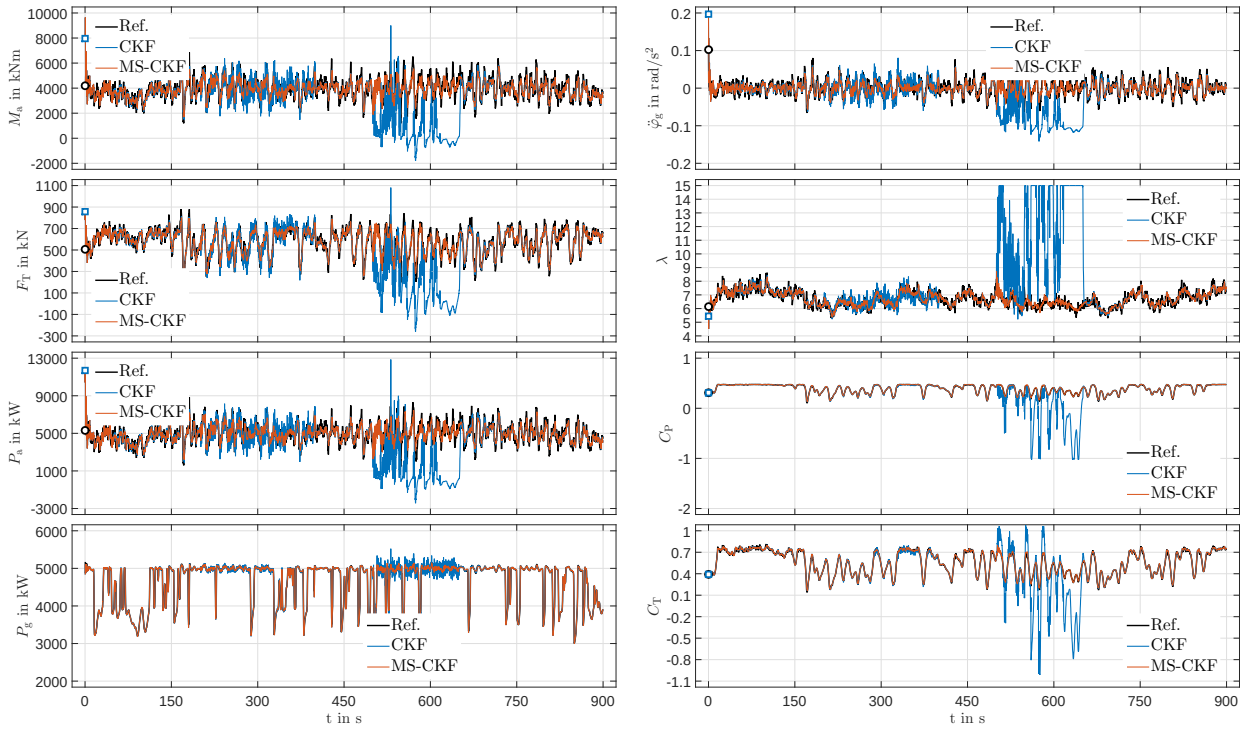

Figure 5.14: Performance variables drawn from the estimation results of the acid test scenario with $v_m = 12\,\mathrm{m/s}$ and TI $= 0.1$ as well as time-varying sensor noise for the complete time window $t \in [0\,,\,900]\,\mathrm{s}$, cf. also Fig. B.11 (p. 155)

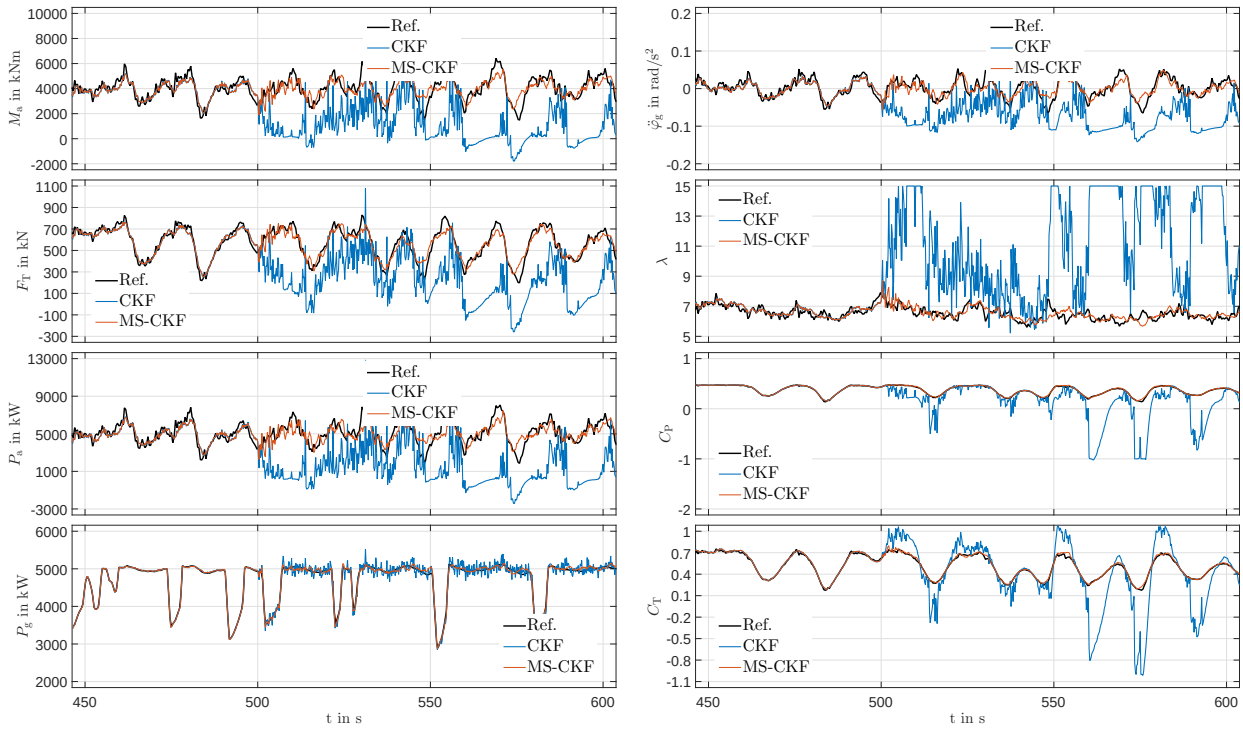

Figure 5.15: Performance variables drawn from the acid test scenario with $v_m = 12\,\mathrm{m/s}$ and TI $= 0.1$ as well as time-varying sensor noise for the time window $t \in [450\,,\,600]\,\mathrm{s})$, cf. also Fig. B.12 (p. 155)

making these estimates more robust in the presence of uncertainties is one major goal for every wind turbine estimator design.

Summarizing, the present section highlights the advantages of the adaptive filter in adverse environments. From a practical point of view, this is an essential step prior to realization on industrial hardware. Additional simulation results, that show two other operating points, are appended in Figs. B.7 to B.10 (p. 153 ff).

5.5 Further Results for Advanced Problems

This section summarizes complementing simulation results of more advanced and adjacent estimation problems which have been addressed, among others, in four accompanying publications [174, 170, 166, 165] (cf. App. C, p. 157). Therein, the following problems have been discussed in particular:

- The distributed state estimation of drive-train and nacelle motion in two directions,
- the advanced state estimation of out-of-plane dynamics with different sensor sets,
- the estimation of process noise using master-slave and Maybeck estimators, and
- the state and load estimation for nonlinear tidal turbine systems.

The purpose of this section is to put these estimation results together into the larger context of the research project and to highlight the added value of the developed model-based virtual sensors.

5.5.1 Distributed State Estimation of Drive-Train and Nacelle Motion

In this section, the advanced estimation problem of simultaneous state and parameter estimation for the wind turbine's drive-train and bi-directional nacelle oscillations is investigated. The following discussion is closely related to the findings in [174].

A straightforward approach is to use a monolithic filter to estimate the states $\boldsymbol{x}$, the unknown parameters $\boldsymbol{\theta}$ and the disturbances $\boldsymbol{d}$. This can be done with an augmented state vector that contains all desired variables. Thereby, the different estimation problems are solved using a joint estimation approach which works fine as long as the augmented system's dimension remains small and practical implementation aspects are excluded. This approach is denoted as monolithic observer architecture, or MOA for short.

As discussed in [174], it is advisable to use a well-composed distributed observer architecture (DOA) rather than a MOA. Fig. 5.16 shows a structural sketch of the proposed architecture.

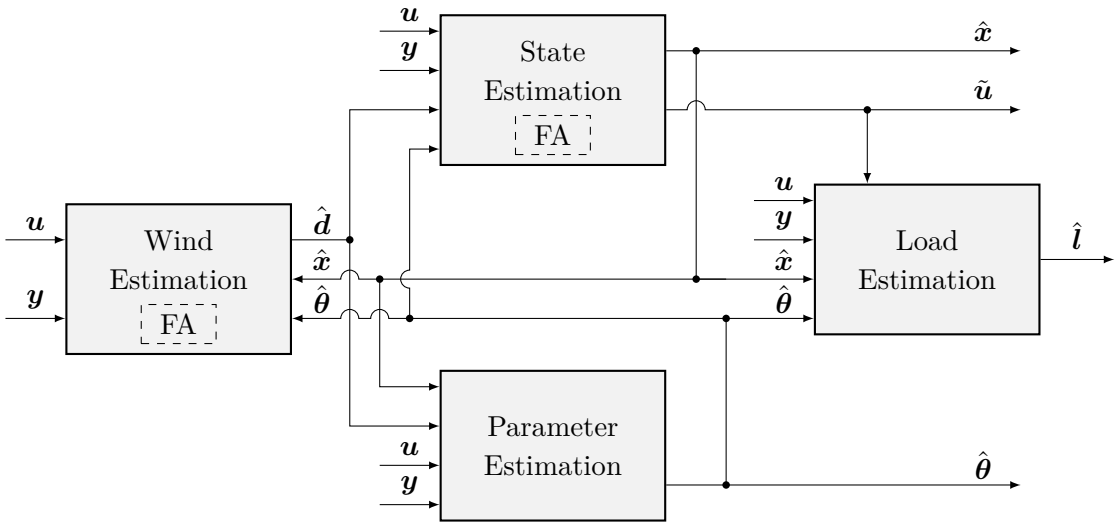

Figure 5.16: Structural sketch of the proposed observer architecture for the distributed wind turbine state and parameter estimation (inspired by [174])

There are four distinct components: a wind estimator (WE), a state estimator (SE), a parameter estimator (PE) and a load estimator (LE). Each of them tackles a different estimation problem with a tailored internal model, filter algorithm and filter parameters.[30]

This filter decomposition makes sense whenever the real-time applicability needs to be considered as one key requirement. The proposed DOA exploits structural properties and negligible internal couplings as well as approximate model linearities (related to the specific wind turbine estimation problem). The internal filter models are derived from the simple and advanced design model equations, presented in Sect. 3.4.4 and 3.4.5.

The exemplary estimation results for a turbulent wind field with $v_\mathrm{m} = 12\,\mathrm{m/s}$ are shown in Fig. 5.17. Therein, MOA and DOA estimates are compared to the true values. From these plots, the following is observed:

- In general, both estimators reconstruct accurately the unknown disturbance input, the states and the mechanical loads. The estimation results produced by MOA and DOA only deviate slightly from one another.
- The wind speed estimation works decently. Fast changes in wind speed (due to high turbulence intensity) are tackled effectively by both estimators even without the nacelle anemometer measurement.
- The average drive-train torsion is observed very accurately, though drive-train oscillations with higher frequency are not observable due to the only roughly known aerodynamic torque on the rotor side. (This can only be improved by additional drive-train sensor equipment like strain gauges).

[30] Please refer to [174] for a more detailed discussion of the distributed observer architecture.

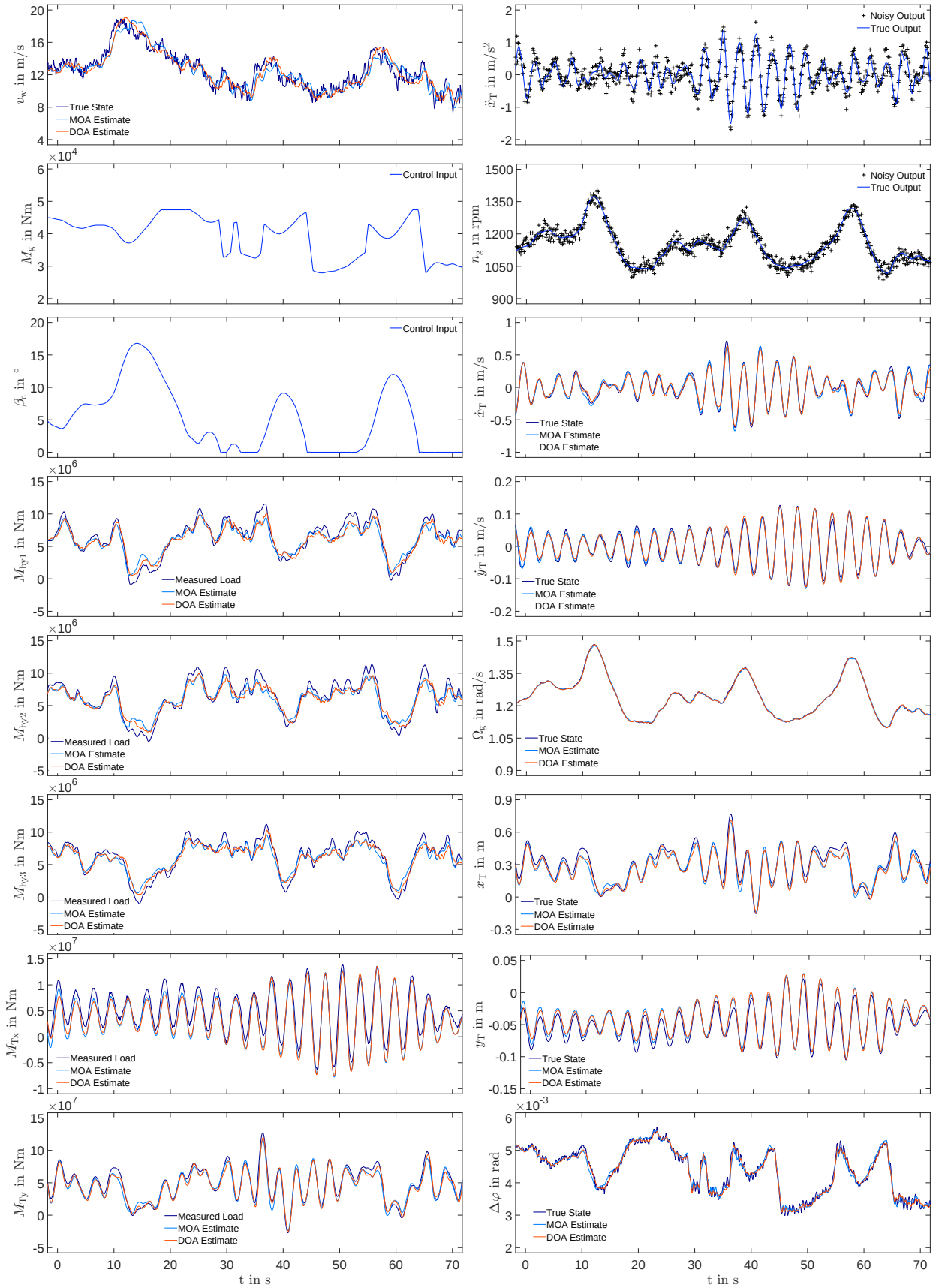

Figure 5.17: Exemplary results obtained for FAST8 simulation data (including medium measurement noise) comparing monolithic and distributed observer architecture [174]

- The employed design models are also useful to reconstruct mechanical loads at the blade root (without installation of blade sensors). A precondition is that the model parameters of blade match the reality precisely which has been assured within this simulation study.
- The plots for the tower-base bending roll and pitch moments $M_{\mathrm{T}x}$ and $M_{\mathrm{T}y}$ show reliable estimation results.

These estimation results indicate a reasonable performance of both nonlinear observer compositions. Despite the corrupted noisy measurements and model errors, the distributed filter architecture provides high-fidelity estimates for the hidden dynamic states.

5.5.2 State Estimation of Rotor Blade Dynamics

Speaking of advanced state estimation, another critical problem poses the online estimation of rotor blade dynamics which has been addressed in a previous publication. Hence, this section recapitulates briefly the relevant results, found and published in [170].

Therein, the focus has been laid on the estimation of blade's out-of-plane, nacelle and drive-train dynamics, considering standard and extended sensor configurations for comparison. As standard sensors, generator speed n_{g} and nacelle accelerations $\ddot{x}_{\mathrm{T}}/\ddot{y}_{\mathrm{T}}$ are considered. The extended configuration also includes the three out-of-plane blade-root bending moments $M_{\mathrm{By},b}$ as (noisy) measurements. The research questions have been:

- Is it possible to reconstruct blade-tip deflections/velocities $x_{\mathrm{B},b}$, $\dot{x}_{\mathrm{B},b}$ without blade measurements? And if, how accurate are the estimates?
- What potential improvement in estimation quality can be expected when considering additional blade-root sensors?

In order to assess these questions, a joint observability and identifiability analysis using empirical Gramian matrices has been conducted (cf. Fig. 5.18). First of all, it was found that most of the states and parameter are well observable with the standard configuration. Only the singular values related to the drive-train torsion and the out-of-plane motion show a lower observability level. The good news is that the extended sensor configuration enhances observability considerably (at least for the latter states), thus suggesting a significant improvement also in estimation quality.

As a second step, a simulation study was conducted where the illustrative estimation results using the FAST8 simulator are shown in Fig. 5.19 and 5.20.

First, Fig. 5.19 presents the investigated test scenario, namely a highly turbulent wind field in the FLR with $v_{\mathrm{m}} = 16\,\mathrm{m/s}$, and illustrates as well the estimation results for drive-train

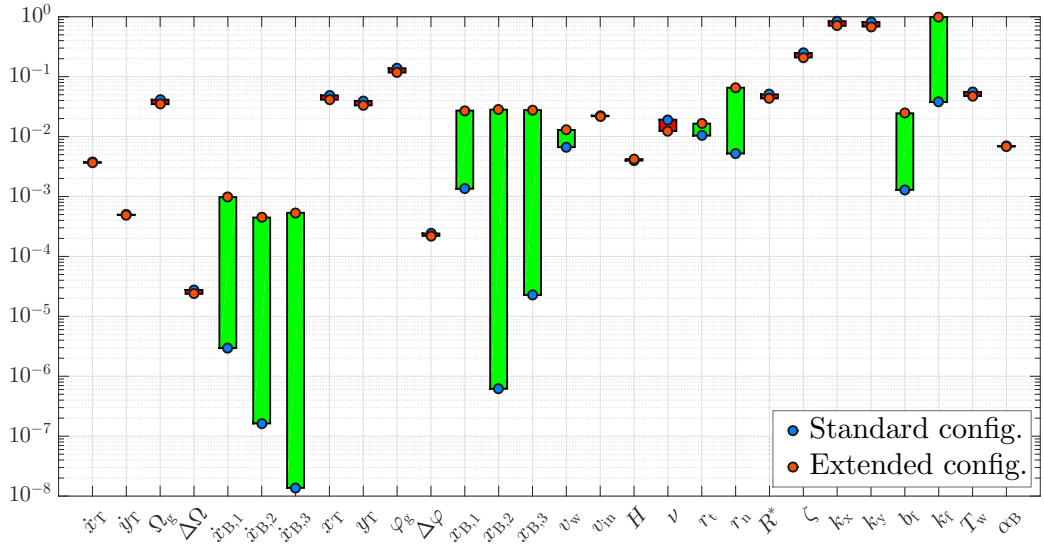

Figure 5.18: Joint observability and identifiability analysis for an advanced wind turbine design model comparing the singular values of the EOG for the standard and extended measurement configuration where the green bars indicate an improvement [170]

and nacelle dynamics. As can be seen from the plots, the estimation quality is already very good. Moreover, virtually no difference between standard and extended sensor configuration is observed (as indicated by the findings of Fig. 5.18). Yet, Fig. 5.20 paints a slightly different picture since the extended sensors outperform the standard ones by far. Despite the difference in estimation accuracy, it is still remarkable that blade-tip deflections are roughly assessable (as observability also predicts).

In conclusion, the focus was laid on the investigation of expectable estimation quality obtained by the standard and the extended configuration. A dedicated observability provided valuable insight about the wind turbine system using a high-order nonlinear design model. The simulation study approved the findings of the observability analysis based on empirical Gramian matrices. According to that investigation, significant improvements from extended configuration arise evidently for the estimation of blade-related states and model parameters.

5.5.3 Process Noise Estimation using MS-CKF and Maybeck's Rule

Another advanced problem constitutes the simultaneous estimation of states and process noise at a time. For this purpose, adaptive filters are needed to estimate also the filter parameters using the same measurement information as the standard filter (cf. Fig. 5.21). The main difference is that there is an additional internal feedback loop using the estimated outputs $\hat{\boldsymbol{y}}_k$ or the states $\hat{\boldsymbol{x}}_k^-/\hat{\boldsymbol{x}}_k^+$. Since there are different possibilities for such automated process noise adaptation, the author has compared two of these concepts, namely the master-slave approach (MS) and the Maybeck estimator (MB), cf. [166].

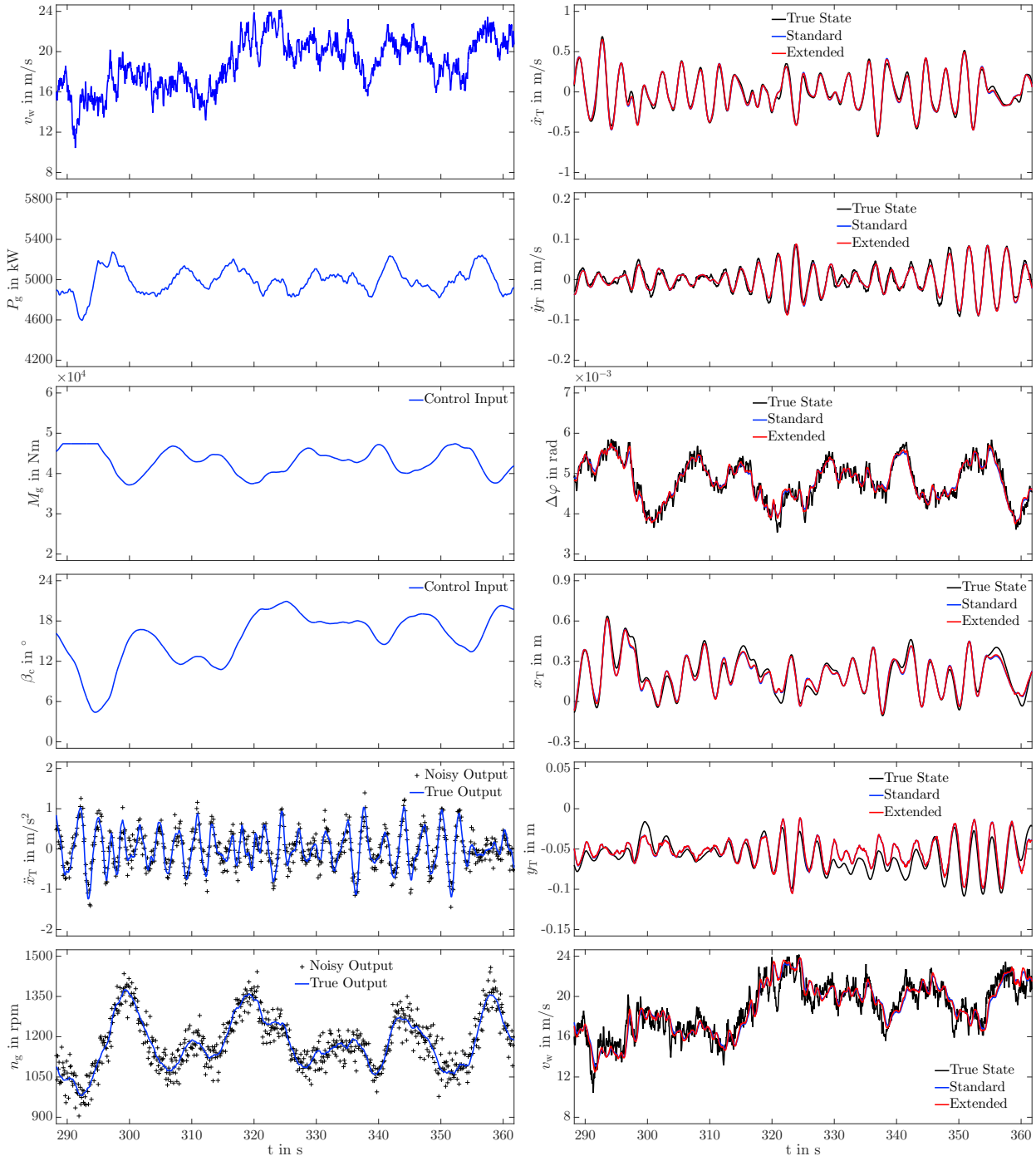

Figure 5.19: Drive-train and nacelle estimates obtained for FAST8 simulation data (including sensor noise) comparing standard and extended measurement configuration [170]

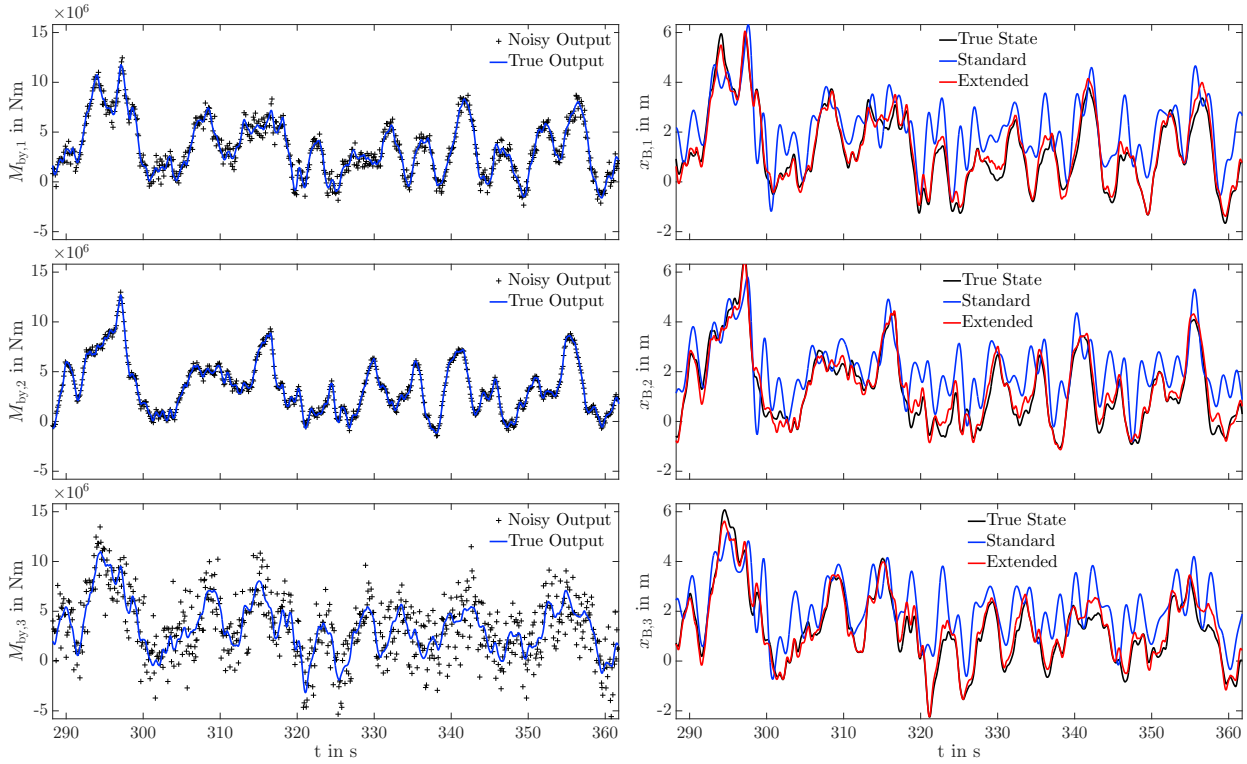

Figure 5.20: Rotor blade estimates obtained for FAST8 simulation data (including sensor noise) comparing standard and extended measurement configuration [170] (left: blade-root bending moments with different noise levels, right: blade-tip deflection)

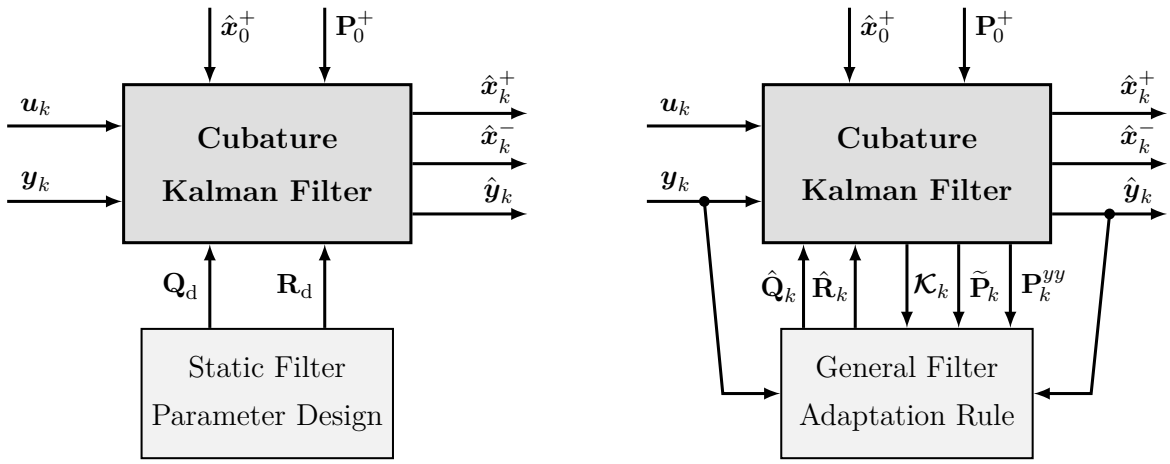

Figure 5.21: Schematic structures of the standard CKF with static filter design (left-hand side) and the noise adaptive CKF with an arbitrary adaptation rule (right-hand side)

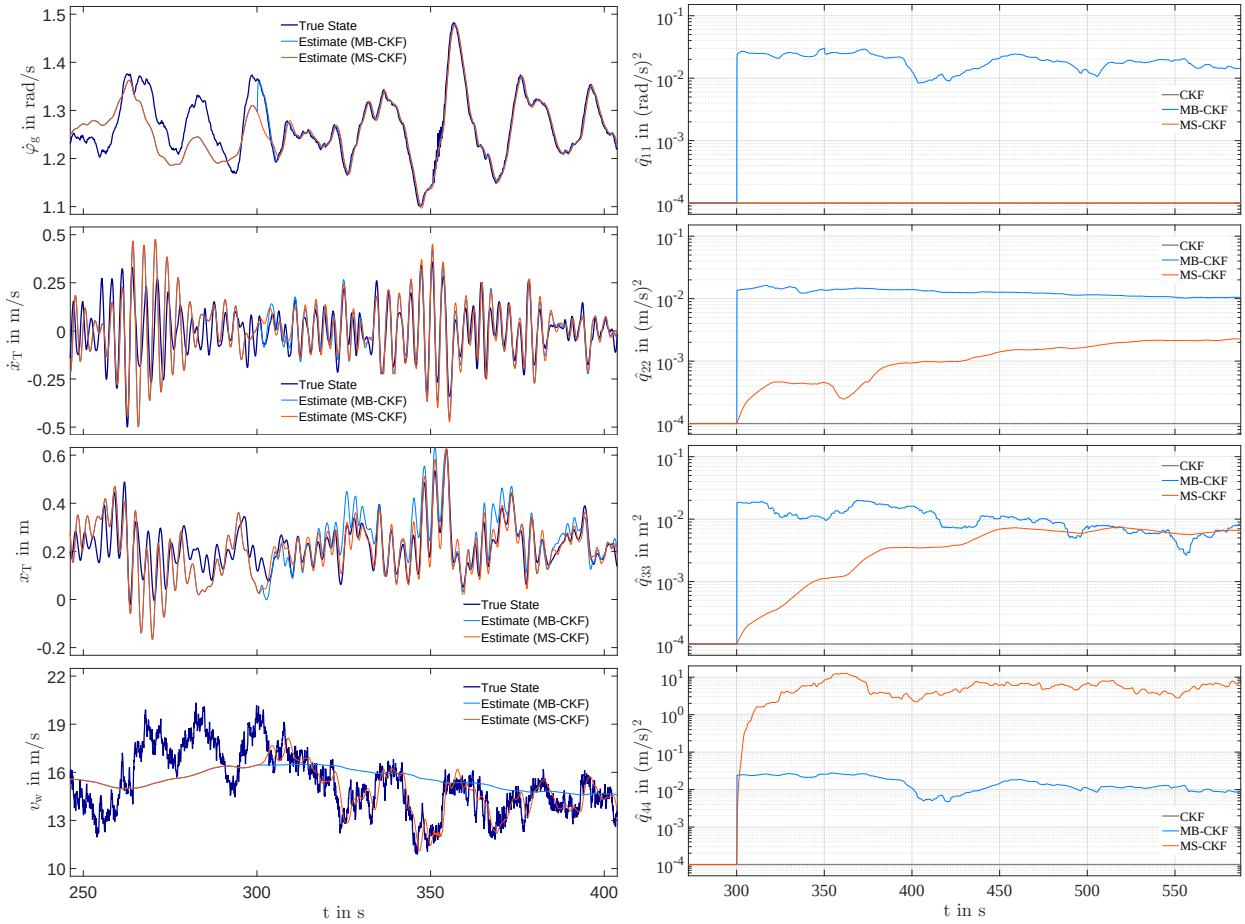

Figure 5.22: Simulation results for process noise adaptation comparing MB-CKF state estimates and MS-CKF with the true states (left-hand side) and the process noise estimates (right-hand side) considering medium input and measurement noise [166]

In particular, the study investigated the effects of a poor initial filter design and the advantages of a dedicated adaptation rule. Fig. 5.22 shows exemplary simulation data comparing MS-CKF and MB-CKF where an arbitrarily poor design has been picked to emphasize the effectiveness of the adaptation. The adaptive filter is initiated at $t = 300\,\mathrm{s}$ and hence before that time all three algorithms produce identical results. Remarkably, both adaptive filters adjust very quickly in order to correct the process noise which is individually perceived as *optimal* by them. Though, differences in estimation results are observed easily.

What does the practitioner take from that? First of all, the algorithm obviously matters. Hence, analyzing available concepts in advance is an important precondition to succeed. Secondly, the initial state also matters since e.g. the MB-CKF adapts the parameters approximately in the same magnitude order. Thus, the approach to optimal filter design must be formulated suitably.

The underlying assumption, why we see such differences, is that there is a lack of observability/identifiability to estimate all eight variables simultaneously which means that there is not enough informational content available. Hence, the standard sensor configuration is

presumably inadequate to facilitate the solution of the full state and (filter) parameter estimation problem. The advantage of the master-slave filter in this context is emphasized such that only the critical parameters are chosen, hence reducing the number of independent variables to be estimated. This point is essential for practical realization.

Consequently, the main conclusions to be drawn from [166] include the following:

- The initial design should be conducted thoroughly and ideally using an algorithmic automated optimization. The more prior information is available, the better the design matches the real noise environment in the field.
- When an adaptation rule needs to be included indispensably, the prior knowledge about which parameters need adjustment is always mission-critical. Hence, the practitioner is encouraged to introduce his or her a priori expert knowledge and pool it conveniently with the automated estimation approach.
- Every concept of filter parameter adaptation has advantages and drawbacks to be considered in the particular application case. Such attributes are computational effort, number of design variables, complexity of the algorithm, suitability for the problem and effort to design the adaptation.

In a nutshell, the above study has revealed different key aspects that have to be considered for the successful application of adaptive filters for controlled wind turbine systems.

5.5.4 Mechanical Load Estimation for Hydro-Kinetic Tidal Turbines

As a short digression, the developed techniques and methodology has been applied to hydro-kinetic turbines as well. This section highlights briefly the findings of a simulation study, recently presented in [165].

The investigated two-bladed tidal turbine has a nominal electrical power $P_0 = 1\,\mathrm{MW}$ and a rotor diameter $D = 21\,\mathrm{m}$ [47]. This reference turbine is conceptualized for a nominal tidal speed $v_0 = 2.5\,\mathrm{m/s}$ and is equipped with active blade pitch control. The control design has been conducted similarly to variable-speed variable-pitch wind turbines (cf. Sect. 3.3.3).

The focus rests in the following on the state estimation in order to predict/reconstruct mechanical loads online or in retrospect: Fig. 5.23 presents the illustrative simulation results for the tidal turbine. Therein, the hydro velocity $v_\mathrm{h}(t)$ as disturbance input, the controlled inputs M_g and β as well as the measurement outputs n_g and $\ddot{x}_\mathrm{T}$ define the investigated test scenario. Moreover, the important state variables and their estimates are compared and several further performance variables are displayed. For load estimation in particular, mainly the hydrodynamic rotor power P_h and rotor torque M_h as well as the rotor thrust force F_T and nacelle reaction force F_R are of greater interest.

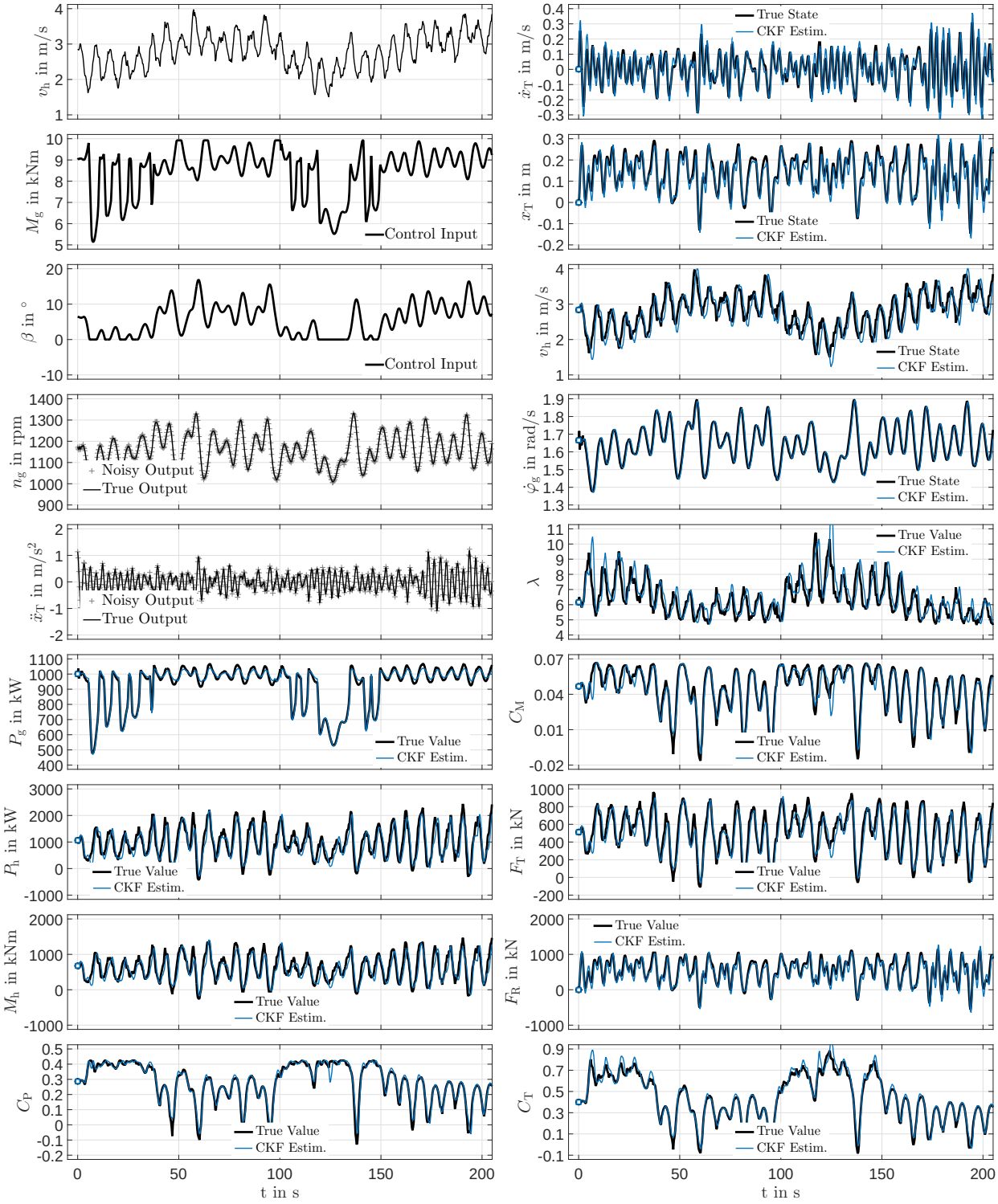

Figure 5.23: Test scenario, state estimates and performance variables obtained for FAST8 simulation data (including sensor noise) comparing the true values (black) to the ones estimated with CKF (light blue) and standard measurement instrumentation [165]

The plots in Fig. 5.23 (p. 132) support the following statements:

- The dynamic states are reconstructed very accurately despite few measurement information available and the unknown, unmeasured tidal velocity.
- This tidal velocity v_{h} itself is also very well estimated. The prevailing effect of the surface interaction, causing larger deterministic velocity fluctuations, is tackled decently and the estimate matches closely the true value.
- The above estimates are the solid foundation to provide highly accurate estimates for critical mechanical loads and other performance variables like tip-speed-ratio λ and power coefficient C_{P}.

In a nutshell, the results demonstrate impressively what is in principle possible to obtain with estimation techniques. Internal states, disturbance inputs and also mechanical loads are estimated in real-time with existing and already installed standard sensors. The employed model-based filter algorithms constitute in principle software-based virtual sensors which play also a key role in advanced control schemes. Thereby, the necessity for supplementary hardware sensors is sidestepped if and when sufficient prior knowledge is available about the dynamic system under investigation. As a prerequisite, the design model needs to be accurately enough.

5.6 Summary and Conclusions

This chapter has presented the most relevant outcomes of the simulation studies conducted to test the designed nonlinear filters. The study has examined various aspects related to which are essential to make these algorithms ready for use in modern control schemes. Furthermore, important results from accompanying publications have been reasoned and put into the greater context.

Since no field data has been directly available at the time of this dissertation project, it has been an important goal to do the investigation under preferably realistic conditions (cf. Sect. 5.3). More advanced problems have been touched in Sect. 5.5 which are further detailed in the accompanying publications of the author [174, 171].[31]

Concluding, the main outcomes that can be drawn from the various simulation studies conducted, are summarized as follows:

- The standard CKF tackles the nonlinear estimation problems very well as long as process and measurement noise are accurately known in advance. Then, the performance of the CKF is comparable to the adaptive MS-CKF.

[31] See also App. C (p. 157) for a complete overview.

- If not, the adaptive slave filter represents an effective and useful extension for the standard SPKF with a moderate increase in computational effort. Unlike the Maybeck estimator, the master-slave approach allows for a selective adaptation of only the critical noise covariances which has been discovered as a key aspect for practical implementation.

- The adaptive CKF shows its superiority in many ways. First, it is capable of addressing unknown and time-varying noise. Secondly, the algorithm is capable to tackle a non-optimal initial filter design to some extent. Thirdly, the user gets additional information about the optimality of the system.

- A dedicated strategy to cope with noise (adaptation) is always necessary. This has been shown illustratively for step changes in measurement noise variances and incorrect initial process covariances.

- As highlighted by several authors before, simultaneous adaptation of process and measurement covariance matrices is not recommended. Particularly, it is not beneficial for filter performance since it makes the filter vulnerable and less robust. A typical problem for adaptive filters occurs whenever the system reaches steady-state conditions or the noise vanishes completely. In such situations, the adaptive filter is prone to divergence which must be avoided from a practical point of view.

- The robust estimation of dynamic states also allows for accurate reconstruction of additional performance variables (cf. Fig. 5.14 and 5.15). Especially the estimates of mechanical component loads can be used as a basis for advanced concepts of predictive maintenance without need for additional sensors.

This chapter concludes the main part of this thesis which consists of the design modeling (Chap. 3), the optimal filter design (Chap. 4) and the performance testing (Chap. 5). The final Chap. 6 closes the circle and provides an overview of the complete thesis, a discussion of its outcomes and eventually a thematic classification in the context of the energy transition.

6 Conclusions and Final Remarks

Wind turbines are one of the largest resources of renewable energy and one of the most promising concepts to replace the prevailing thermal power plants in future. For this to happen, the scientific and industrial community must work together to improve steadily the performance of next-generation wind turbines and thereby create the economic benefits needed to establish wind turbine technology. This thesis is intended to provide a small jigsaw piece to the overall development. Having said this, it is finally important to assess how the obtained outcomes of this thesis might have an effect regarding the bigger picture and the ongoing energy transition.

6.1 Concluding Summary

This thesis has dealt with nonlinear and adaptive state estimation techniques for wind turbine application and control. This topic still constitutes a niche in the wind energy research community and has not been treated with the necessary profoundness yet (unlike i.e. wind turbine control).

The **first chapter** (p. 1 ff) has provided an introduction to the research field and has formulated the primary research objectives. Moreover, a complete literature overview has been given on state estimation for wind turbine application. This overview has revealed that research on certain aspects of state estimation has been accomplished in the past, though no complete treatment exists to this day.

The **second chapter** (p. 15 ff) has introduced the necessary background knowledge on Kalman filtering in a concise fashion. The algorithms for the standard and adaptive Kalman filters as well as for linear and nonlinear systems are made available such that everything needed for state estimation is included in this thesis. The research has concentrated early on the sigma-point approach. This decision was made due to several beneficial properties of the SPKF. The most relevant ones are the superior numerical properties, the moderate computational effort required for real-time application and the ease of implementation. Adaptive filters are included as a necessary add-on in order to manage the problem of unknown and time-varying noise covariances of the Kalman filter. The final choice for the master-slave concept offered additional benefits. The practicability is increased due to the fact that the same algorithm can be used for the master and slave filter. This means that the same design methodology applies and both filters have the same numerical properties. Moreover, the expenditure on testing, implementation and evidence of functionality is reduced.

The **third chapter** (p. 43 ff) introduces the wind turbine control system under investigation. The basics of the wind energy conversion are presented, the relevant components from a control engineers perspective are described. Moreover, this chapter makes the control-oriented models available which are needed by the estimators as filter models. Several models with different granularities are derived which are used selectively depending on the actual estimation problem. Thereby, the design model is always tailored to the specific estimation problem and a model library has been set up to switch deliberately between these models.

The **fourth chapter** (p. 81 ff) proposes a novel methodology for filter parameter design for wind turbine state estimation. This methodology applies numerical optimization techniques in order to replace the commonly used trial-and-error method. This approach allows for a highly automated filter design in practise which eases the design effort considerably. It has been shown that this approach works quite well. Two different objective functions have been investigated. The first one uses manually chosen weighting matrices to put emphasize on certain states or outputs. The second function is based on the expected consistency of the estimator.

The **fifth chapter** (p. 105 ff) consolidates the findings of the previous chapters. The proposed sigma-point filters are tested with and without noise adaptation using various test scenarios. Among others, the acid test scenario is applied which highlights the superiority of the adaptive filter if noise statistics are unknown and time-varying. Additionally, the chapter has gathered the findings of four different aspects of state estimation from four different accompanying publications. These were the proposed distributed observer architecture, the estimation of blade loads, the process noise adaptation and the load estimation for tidal turbines.

6.2 Main Contributions

This thesis has conducted a profound investigation of nonlinear state estimation techniques for wind turbine application. In this regard, the thesis takes a closer look at the critical aspects when bringing state estimation to successful practical application in the field. These aspects are the selection and evaluation of suitable filter algorithms, the target-oriented physical modeling, the filter design methodology and, finally, the acid testing with a thorough simulation study. The key challenges have been addressed and investigated in this thesis with a comprehensive simulation study. Additional results were published previously in [167, 174, 170, 166, 169, 171].

To sum up, the following three contributions are considered as the most relevant ones:

1. Derivation of target-oriented models and improvement of existing nonlinear filters

   The first part of this thesis puts emphasis on the profound choice of a tailored algorithm that is well suited to the real-time requirements of industrial controllers, that is also

robust against poor parametrization and has also beneficial numerical properties. The investigation has revealed the master-slave adaptive SPKF as the most suitable one. Moreover, the thesis derives a number of physical models which are employed as part of the specific filter realization. These models are used for different estimation purposes such as wind speed estimation, state estimation and mechanical load estimation.

2. Development of a novel filter design methodology

   As a second contribution, this thesis has introduced a novel filter design approach for wind turbines. Thereby, the typical manual design approach is completely replaced by the automated solution of a well-defined optimization problem. This so-called automated design is highly preferable in order to reduce the manual intervention to a minimum. Two promising objective functions have been introduced and compared using either the weighted or the normalized estimation error squared. The solution of the optimization problems is found effectively by interior-point methods.

3. Proof of concept for the proposed nonlinear and adaptive filters

   The third contribution of this thesis is the simulative proof of concept for the investigated algorithms. Various test scenarios for different estimation problems have been conducted successfully. The final acid testing under adverse practical conditions showed in particular the superiority of the adaptive filter for unknown and time-varying noise statistics. Moreover, the combined state and mechanical load estimation has been demonstrated for tower and blade loads with and without advanced sensors. Finally, the complete design methodology has been transferred illustratively to hydro-kinetic tidal turbines.

6.3 Outlook and Future Work

This final section places the findings of the thesis in the bigger picture of energy transition and presents then possible pathways for future research related to state estimation.

The Bigger Picture

The anthropogenic climate change is real. It happens even more rapidly than widely expected or, more accurately speaking, than it was hoped for by many. Mankind has become a climatic factor long ago and now the consequences have become dire. Only little time is left to avoid the most disastrous impacts on our everyday life. The sensible pathway to tackle climate change would be to bring the net greenhouse gas emissions down [157]. Disastrously, half of the man-made CO_2 in the atmosphere today was emitted after the United Nations Earth Summit in Rio de Janeiro in 1992 where global warming came into focus of the general public. Though this knowledge did not lead to actions that resulted in lower emissions. Therefore today less political statements of intent are needed, but more unprecedented efforts.

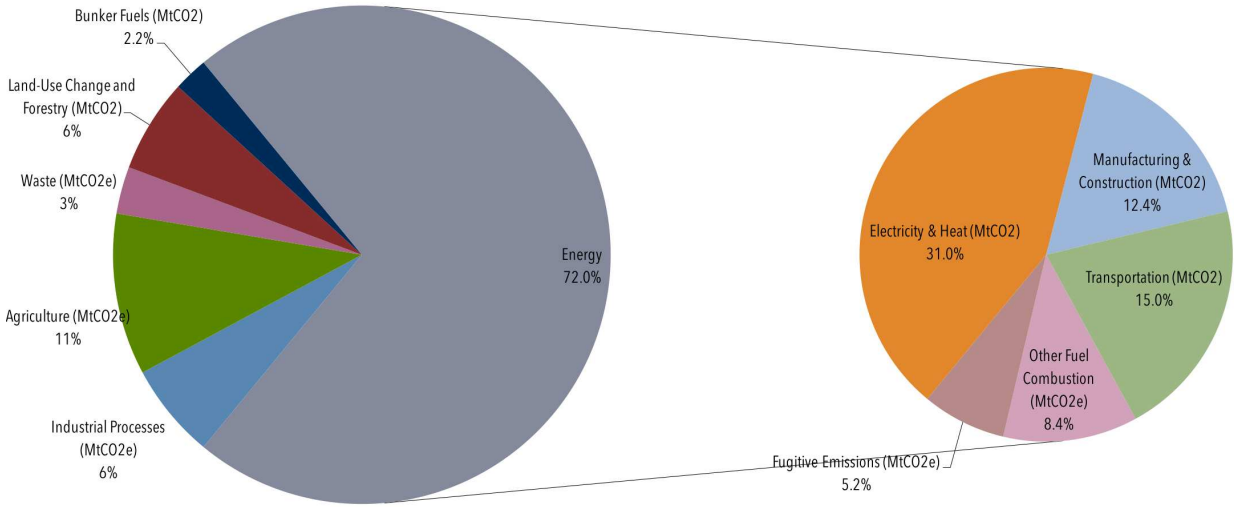

Figure 6.1: Globally, the primary sources of greenhouse gas emissions are electricity and heat (31%), agriculture (11%), transportation (15%), forestry (6%) and manufacturing (12%). Energy production of all types accounts for 72 percent of all emissions. Source: Climate Analysis Indicators Tool (World Resources Institute, 2017). https://www.c2es.org/[...] [accessed 2020-01-16]

Since the electrical energy consumption from fossil fuels has a substantial share of the global CO_2 emissions (cf. Fig. 6.1), this constitutes an effective lever to reduce emissions. From a technological perspective, wind turbines (in combination with solar plants) are the right pathway with a comparatively small carbon footprint (cf. Fig. 1.1). The development has been a veritable success story since today's onshore turbines have become competitive and are already the largest contributor of renewable energy. Admittedly, this possibility to reduce emissions effectively seems not to be very likely today because earth population is growing, deforestation and environmental destruction are rising worldwide, global consumption is increasing and thus also CO_2 emissions are still on a steep ascent (cf. Fig. 6.2) although they should actually be reduced. The situation is already so dramatic, even renowned scientists tend to say that *it is time to panic* now [157].

Nevertheless, even if the situation appears hopeless today, resignation is no real alternative. The struggle for a renewable energy future has just begun. If the consequences of climate change have become so harsh for everybody (also in Europe) then the necessary political will, the necessary budget and the economic interests must be overwhelming eventually. Unfortunately, then it does not suffice to reduce emissions only but active measures of carbon storages and also solar geoengineering are possibly inevitable[32] which will be a *technologically possible, economically feasible, and potentially politically disruptive way* for state as well as non-state actors [163]. Though it should be clear from the start, that this path is a very dangerous one to follow.

[32] Gernot Wagner & Martin L. Weitzman: *A Big-Sky Plan to Cool the Planet*, The Wall Street Journal, https://gwagner.com/[...] [accessed 2019-09-03]

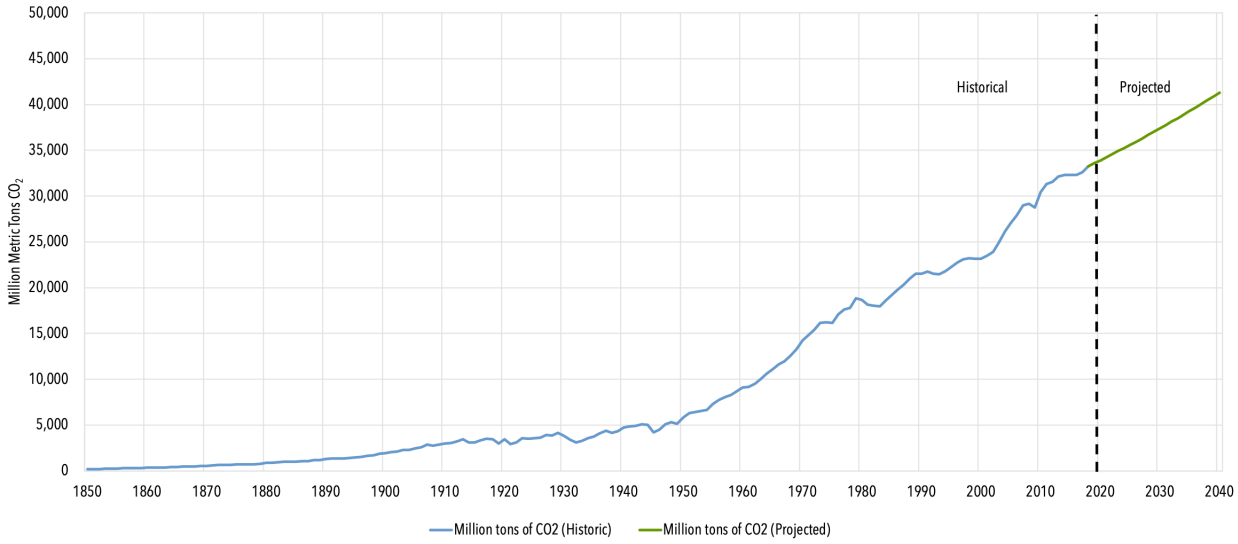

Figure 6.2: Global Carbon Dioxide Emissions, 1850–2040. Source: Carbon Dioxide Information Analysis Center (Oak Ridge National Laboratory, 2017), World Energy Outlook (International Energy Agency, 2019). https://www.c2es.org/[...] [accessed 2020-01-16]

In order to build the bridge to this thesis, new technological improvements are needed to strengthen the market penetration of wind turbines. Advanced control solutions offer improvements for both, the operator (reduced LCoE and life-time extension) and the manufacturer (reduced expenses for new controller development). The contributions on state estimation, developed in this thesis, provide the necessary foundation to bring virtual sensors successfully into application with these control solutions. Together, state estimation and advanced control are one brick in the wall to more cost-effective turbines which helps wind energy to persist in the market and outperform other technologies hopefully quickly enough (cf. Fig. 1.1, p. 2).

In summary, taking action is better than resignation. Therefore, the next paragraph suggests future paths for research in the field of state estimation and virtual sensing for wind turbines as well as closely related topics.

Recommendations for Future Work

Eventually, this industry-oriented research has revealed several aspects that have not been treated in detail in this thesis since these were out of the scope. These possible directions for future work in the field of state estimation are highlighted hereafter:

- Investigation with field test data

  The designed and implemented algorithms have not yet been tested with real measurement data since no field data was available at the time of this dissertation project. This is a minor weak point since the performance has already been validated with realistic simulation data from a high-fidelity simulator. Still, it will be necessary to

investigate the filter performance also under real conditions and perform a proof of concept on a real wind turbine.

- Investigation with advanced control concepts

  Advanced wind turbine control has not been in the focus of this thesis. Thus, also load reducing control strategies have not been included. Closing the control loop with a state feedback will be the next logical step. It should be investigated how the controller and state estimator work together in the closed loop and how large the difference in control performance is compared to a controller that has ground-truth information about the states.

- Identifiability analysis for filter parameters

  As discussed in this thesis, the adaptation of noise parameters can be regarded as a special case of an identification problem. In order to estimate the filter parameters, the system must be sufficiently excited (as precondition) and the filter parameters must be identifiable in principle. Practically, it has been observed for the adaptive filter design that certain noise parameters (primarily for process noise) cannot be estimated together robustly. There is thus reason to presume that this is related directly to a lack of identifiability. For better clarity, a systematic investigation should take a closer look at this topic.

- Concepts for monitoring and predictive maintenance

  The developed filters provide also estimates of relevant structural loads of the wind turbine without direct measurement. If these estimates are logged continuously, the time history information is available over months or even years. Such data could be analysed online to estimate the remaining life time of critical components (e.g. main bearing, gear-box, tower, rotor blades etc.). Moreover, this information can be exploited for the purpose of predictive maintenance to avoid unplanned shut-downs. Future work needs to develop fatigue load models for the critical components in order to convert the structural loads directly to effective component wear. Such a model could be part of another filter that predicts the current wear states of the wind turbine.

A Filter Algorithms

This appendix presents some mathematical preliminaries required for understanding the SPKF algorithms. Moreover, the classical and square-root UKF filter are introduced and the slave cubature Kalman filter and the derivation of the linear slave KF is provided.

A.1 Mathematical Preliminaries

A.1.1 Cholesky Decomposition

The Cholesky decomposition (or factorization) is mathematical concept used to decompose a Hermitian, positive definite matrix $\mathbf{A} \in \mathbb{C}^{n \times n}$, defined by

$$\mathbf{A} = \mathbf{L}\,\mathbf{L}^{*}\,, \tag{A.1}$$

into a lower triangular matrix $\mathbf{L}$ and its conjugate transpose $\mathbf{L}^{*}$ [161] (p. 100f). Note that, $\mathbf{A} = \mathbf{A}^{*}$ is to be presumed (since $\mathbf{A}$ being Hermitian). The matrix $\mathbf{L}$ is denoted as the so-called Cholesky factor and constitutes the generalization of matrix square-roots. The decomposition in Eq. (A.1) is unique if $\mathbf{A}$ is positive definite. If $\mathbf{A}$ is a real matrix, the Eq. (A.1) turns into

$$\mathbf{A} = \mathbf{L}\,\mathbf{L}^{\mathrm{T}} \tag{A.2}$$

where $\mathbf{A} = \mathbf{A}^{\mathrm{T}}$ holds. The Cholesky factor is also depicted as $\mathbf{L} = \mathrm{chol}\{\mathbf{A}\}^{\mathrm{T}}$ or simply $\mathbf{L} = \sqrt{\mathbf{A}}$. The Cholesky decomposition can be used to solve systems of linear equations, to solve optimization problems or to improve filter algorithms, among others, exploiting the properties of the matrix $\mathbf{A}$ in an efficient manner.

In the context of Kalman filters, the Cholesky factor is labelled in this thesis using the variable $\mathbf{S}$, indicating the square-root of the covariance matrix.

A.1.2 QR Decomposition

The QR decomposition/factorization is a well-known and frequently applied mathematical routine [54] that is used to decompose a rectangular matrix $\mathbf{A} \in \mathbb{R}^{n \times m}$, defined by

$$\mathbf{A}^{\mathrm{T}} = \mathbf{Q}\mathbf{R}\,, \tag{A.3}$$

into an unitary/orthonormal matrix $\mathbf{Q} \in \mathbb{R}^{n \times n}$ and an upper-triangular matrix $\mathbf{R} \in \mathbb{R}^{m \times n}$ where $m \geq n$ is assumed. Note that, $\mathbf{Q}^{\mathrm{T}}\mathbf{Q} = \mathbf{I}$ and thus $\mathbf{Q}^{\mathrm{T}} = \mathbf{Q}^{-1}$ holds. The computation

can be done in several ways, among others the Householder transformation, the Givens transformation or Gram-Schmidt orthogonalization.

The Kalman filter's covariance matrix can be depicted as a matrix product

$$\mathbf{P} = \mathbf{A}\mathbf{A}^{\mathrm{T}} \tag{A.4}$$

wherein the matrix $\mathbf{A}$ could be interpreted as the matrix square-root (MSR) of the matrix $\mathbf{P}$, though not being an upper triangular matrix per se. However, this is not preferable from a numerical point of view in general. Consider the following example with

$$\mathbf{A} = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix} \qquad \text{and} \qquad \mathbf{S} = \begin{bmatrix} \sqrt{2} & 0 \\ \sqrt{2}/2 & \sqrt{6}/2 \end{bmatrix}. \tag{A.5}$$

for the purpose of illustration. As can be easily comprehended, both matrices yield the same matrix product

$$\begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} = \begin{bmatrix} \sqrt{2} & 0 \\ \sqrt{2}/2 & \sqrt{6}/2 \end{bmatrix} \begin{bmatrix} \sqrt{2} & \sqrt{2}/2 \\ 0 & \sqrt{6}/2 \end{bmatrix} \tag{A.6}$$

In order to obtain the desired Cholesky factor $\mathbf{S} \in \mathbb{R}^{n\times n}$ directly (as a square upper-triangular matrix) from an arbitrary-sized and given matrix $\mathbf{A}$, the QR decomposition is exploited such that

$$\mathbf{P} = \mathbf{A}\mathbf{A}^{\mathrm{T}} = (\mathbf{Q}\mathbf{R})^{\mathrm{T}}\mathbf{Q}\mathbf{R} = \mathbf{R}^{\mathrm{T}}\mathbf{Q}^{\mathrm{T}}\mathbf{Q}\mathbf{R} = \mathbf{R}^{\mathrm{T}}\mathbf{R} = \mathbf{S}\,\mathbf{S}^{\mathrm{T}} \tag{A.7}$$

is gained. This has been derived by inserting Eq. (A.3) into Eq. (A.4) and considering the properties of $\mathbf{Q}$. The Cholesky factor $\mathbf{S}$ is unique if $\mathbf{P} \in \mathbb{R}^{n\times n}$ is symmetric positive definite [54]. That is, the assessment of the matrix square-root (e.g. in the SPKF) can be readily replaced by a simple QR decomposition. In general, $\mathbf{R}$ is upper-triangular but not square for sure (in case of $m > n$). Nevertheless, its upper part $\mathbf{U} \in \mathbb{R}^{n\times n}$ still contains all the necessary information and, hence, can be directly employed since both

$$\mathbf{U}^{\mathrm{T}} = \mathbf{S} \tag{A.8}$$

and also $\mathbf{R}^{\mathrm{T}}\mathbf{R} = \mathbf{U}^{\mathrm{T}}\mathbf{U} = \mathbf{P}$ hold true. In the above example, the QR factorization for $\mathbf{A}^{\mathrm{T}}$ yields

$$\mathbf{Q} = \begin{bmatrix} \sqrt{2}/2 & -\sqrt{6}/6 & -\sqrt{3}/3 \\ \sqrt{2}/2 & \sqrt{6}/6 & \sqrt{3}/3 \\ 0 & \sqrt{6}/3 & -\sqrt{3}/3 \end{bmatrix} \qquad \text{and} \qquad \mathbf{R} = \begin{bmatrix} \sqrt{2} & \sqrt{2}/2 \\ 0 & \sqrt{6}/2 \\ 0 & 0 \end{bmatrix} \tag{A.9}$$

and thus this result (obviously) agrees with Eq. (A.5) and Eq. (A.3). The above relationships are taken advantage of in the square-root evaluation of Algorithm 2.4 and in the following Sect. A.3.

A.1.3 Singular Value Decomposition

The singular value decomposition, or SVD for short, is the third (important) factorization concept of linear algebra considered in this thesis (cf. [206], p. 331 ff). For start, consider an arbitrary matrix $\mathbf{A} \in \mathbb{C}^{m \times n}$. Then, the SVD decomposes the matrix such that

$$\mathbf{A} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^* \tag{A.10}$$

is obtained. Note that $\mathbf{U} \in \mathbb{C}^{m \times m}$ and $\mathbf{V} \in \mathbb{C}^{n \times n}$ are unitary matrices where $\mathbf{U}^*\mathbf{U} = \mathbf{I}$, $\mathbf{U}^* = \mathbf{U}^{-1}$, $\mathbf{V}^*\mathbf{V} = \mathbf{I}$ and $\mathbf{V}^* = \mathbf{V}^{-1}$ hold true. The matrix $\boldsymbol{\Sigma} \in \mathbb{R}^{m \times n}$ is a real matrix containing the singular values σ_i for $i = 1, 2, \ldots, n$ with σ_n being the smallest singular value and σ_1 the largest one.

These two singular values are employed to assess the inverse condition number (ICN) defined by

$$\mathrm{ICN}(\mathbf{A}) = \mathrm{cond}\{\mathbf{A}\}^{-1} = \frac{\sigma_n(\mathbf{A})}{\sigma_1(\mathbf{A})} \in [0, 1] \tag{A.11}$$

that is employed, for instance, as a measure for the degree of observability of the system (whereby a value closer to 1 indicates a higher observability).

In case of a real matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$, Eq. (A.10) becomes

$$\mathbf{A} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^{\mathrm{T}} \tag{A.12}$$

because the (above) conjugate transposes are replaced by simple transposes. Moreover, in case of $\mathbf{A}$ also being square and symmetric positive definite, Eq. (A.12) further reduces to

$$\mathbf{A} = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^{\mathrm{T}} \tag{A.13}$$

where $\boldsymbol{\Lambda}$ is the square matrix containing the eigenvalues λ_i of $\mathbf{A}$. Refer to [206] for further details on the SVD.

A.1.4 Sum of Dyadic Products

The relationship between the sum over a set of n dyadic vector products and a simple matrix operation is constituted as follows:

$$\sum_{i=1}^{n} \boldsymbol{a}_i \boldsymbol{a}_i^{\mathrm{T}} = \boldsymbol{a}_1 \boldsymbol{a}_1^{\mathrm{T}} + \boldsymbol{a}_2 \boldsymbol{a}_2^{\mathrm{T}} + \ldots + \boldsymbol{a}_n \boldsymbol{a}_n^{\mathrm{T}} \tag{A.14}$$

$$= \begin{bmatrix} \boldsymbol{a}_1 & \boldsymbol{a}_2 & \cdots & \boldsymbol{a}_n \end{bmatrix} \begin{bmatrix} \boldsymbol{a}_1^{\mathrm{T}} \\ \boldsymbol{a}_2^{\mathrm{T}} \\ \vdots \\ \boldsymbol{a}_n^{\mathrm{T}} \end{bmatrix} \tag{A.15}$$

$$= \begin{bmatrix} \boldsymbol{a}_1 & \boldsymbol{a}_2 & \cdots & \boldsymbol{a}_n \end{bmatrix} \begin{bmatrix} (\boldsymbol{a}_1^{\mathrm{T}})^{\mathrm{T}} & (\boldsymbol{a}_2^{\mathrm{T}})^{\mathrm{T}} & \cdots & (\boldsymbol{a}_n^{\mathrm{T}})^{\mathrm{T}} \end{bmatrix}^{\mathrm{T}} \tag{A.16}$$

$$= \begin{bmatrix} \boldsymbol{a}_1 & \boldsymbol{a}_2 & \cdots & \boldsymbol{a}_n \end{bmatrix} \begin{bmatrix} \boldsymbol{a}_1 & \boldsymbol{a}_2 & \cdots & \boldsymbol{a}_n \end{bmatrix}^{\mathrm{T}} \tag{A.17}$$

$$= \mathbf{A}\mathbf{A}^{\mathrm{T}} \tag{A.18}$$

A.2 The Standard Unscented Kalman Filter

The standard version of the UKF using the scaled unscented transformation (SUT) with the scaling parameter α, the parameter β for the higher order moments and κ for the skew (cf. [132], p. 56) is provided in Algorithm A.1. Note that, the matrix square roots of $\mathbf{P}_{k-1}^{+}$ and $\mathbf{P}_{k}^{-}$ are obtained in every recursion step by the Cholesky decomposition, cf. App. A.1.1.

Algorithm A.1 (The Standard Unscented Kalman Filter)

Initialization step:

Initial state mean
$$\hat{\boldsymbol{x}}_0^{+} = E\{\boldsymbol{x}_0\} \tag{A.19a}$$

Initial covariance
$$\mathbf{P}_0^{+} = E\Big\{(\boldsymbol{x}_0 - \hat{\boldsymbol{x}}_0)(\boldsymbol{x}_0 - \hat{\boldsymbol{x}}_0)^{\mathrm{T}}\Big\} \tag{A.19b}$$

Sigma-point weights
$$\lambda = \alpha^2(n_x + \kappa) - n_x\,,\ w_{\mathrm{p}} = \sqrt{n_x + \lambda} \tag{A.19c}$$

$i = 0 \Rightarrow$
$$w_{\mathrm{m},0} = \frac{\lambda}{n_x + \lambda},\ w_{\mathrm{c},0} = \frac{\lambda}{n_x + \lambda} + (1 - \alpha^2 + \beta) \tag{A.19d}$$

$i = 1 \ldots 2n_x \Rightarrow$
$$w_{\mathrm{m},i} = \frac{1}{2(n_x + \lambda)},\ w_{\mathrm{c},i} = \frac{1}{2(n_x + \lambda)} \tag{A.19e}$$

Prediction step:

New sigma-points
$$\boldsymbol{\mathcal{X}}_{k-1}^{+} = \widehat{\mathbf{X}}_{k-1}^{+} + w_{\mathrm{p}}\left[0, \sqrt{\mathbf{P}_{k-1}^{+}}, -\sqrt{\mathbf{P}_{k-1}^{+}}\right] \tag{A.19f}$$

Propagated sigma-points
$$\boldsymbol{\mathcal{X}}_{k}^{*} = \boldsymbol{f}(\boldsymbol{\mathcal{X}}_{k-1}^{+}, \boldsymbol{u}_{k-1}) \tag{A.19g}$$

Predicted state estimate
$$\hat{\boldsymbol{x}}_k^{-} = \sum_{i=0}^{2n_x} w_{\mathrm{m},i}\, \boldsymbol{\mathcal{X}}_k^{*}(:,i) \tag{A.19h}$$

Predicted covariance matrix
$$\mathbf{P}_k^{-} = \sum_{i=0}^{2n_x} w_{\mathrm{c},i}\Big(\boldsymbol{\mathcal{X}}_k^{*}(:,i) - \hat{\boldsymbol{x}}_k^{-}\Big)\Big(\boldsymbol{\mathcal{X}}_k^{*}(:,i) - \hat{\boldsymbol{x}}_k^{-}\Big)^{\mathrm{T}} + \mathbf{Q}_{k-1} \tag{A.19i}$$

Correction step:

New sigma-points
$$\boldsymbol{\mathcal{X}}_{k}^{-} = \widehat{\mathbf{X}}_{k}^{-} + w_{\mathrm{p}}\left[0, \sqrt{\mathbf{P}_{k}^{-}}, -\sqrt{\mathbf{P}_{k}^{-}}\right] \tag{A.19j}$$

Propagated sigma-points
$$\boldsymbol{\mathcal{Y}}_{k}^{*} = \boldsymbol{h}(\boldsymbol{\mathcal{X}}_{k}^{-}, \boldsymbol{u}_{k}) \tag{A.19k}$$

Predicted output
$$\hat{\boldsymbol{y}}_k = \sum_{i=0}^{2n_x} w_{\mathrm{m},i}\, \boldsymbol{\mathcal{Y}}_k^{*}(:,i) \tag{A.19l}$$

Cross covariance matrix
$$\mathbf{P}_k^{xy} = \sum_{i=0}^{2n_x} w_{\mathrm{c},i}\Big(\boldsymbol{\mathcal{X}}_k^{*}(:,i) - \hat{\boldsymbol{x}}_k^{-}\Big)\Big(\boldsymbol{\mathcal{Y}}_k^{*}(:,i) - \hat{\boldsymbol{y}}_k\Big)^{\mathrm{T}} \tag{A.19m}$$

Innovation covariance matrix
$$\mathbf{P}_k^{yy} = \sum_{i=0}^{2n_x} w_{\mathrm{c},i}\Big(\boldsymbol{\mathcal{Y}}_k^{*}(:,i) - \hat{\boldsymbol{y}}_k\Big)\Big(\boldsymbol{\mathcal{Y}}_k^{*}(:,i) - \hat{\boldsymbol{y}}_k\Big)^{\mathrm{T}} + \mathbf{R}_k \tag{A.19n}$$

Kalman Gain
$$\boldsymbol{\mathcal{K}}_k = \mathbf{P}_k^{xy}(\mathbf{P}_k^{yy})^{-1} \tag{A.19o}$$

Corrected covariance matrix
$$\mathbf{P}_k^{+} = \mathbf{P}_k^{-} - \boldsymbol{\mathcal{K}}_k \mathbf{P}_k^{yy} \boldsymbol{\mathcal{K}}_k^{\mathrm{T}} \tag{A.19p}$$

Filter Innovation
$$\boldsymbol{v}_k = \boldsymbol{y}_k - \hat{\boldsymbol{y}}_k^{-} \tag{A.19q}$$

Corrected state estimate
$$\hat{\boldsymbol{x}}_k^{+} = \hat{\boldsymbol{x}}_k^{-} + \boldsymbol{\mathcal{K}}_k \boldsymbol{v}_k \tag{A.19r}$$

A.3 The Square-Root Unscented Kalman Filter

The square-root version of the UKF is provided in Algorithm A.2. Note that in Eq. (A.20r) the Cholesky update is due to the -1 effectively a downdate on the corrected covariance matrix square-root. Furthermore, $\mathbf{S}_k^{\mathrm{Q}} = \mathrm{chol}\{\mathbf{Q}_k\}^{\mathrm{T}}$ and $\mathbf{S}_k^{\mathrm{R}} = \mathrm{chol}\{\mathbf{R}_k\}^{\mathrm{T}}$ hold.

Algorithm A.2 (The Square-Root Unscented Kalman Filter)

Initialization step:

$$\text{Initial state mean} \qquad \hat{\boldsymbol{x}}_0^+ = E\{\boldsymbol{x}_0\} \tag{A.20a}$$

$$\text{Initial covariance} \qquad \mathbf{S}_0^+ = \mathrm{chol}\Big\{E\Big\{(\boldsymbol{x}_0 - \hat{\boldsymbol{x}}_0)(\boldsymbol{x}_0 - \hat{\boldsymbol{x}}_0)^{\mathrm{T}}\Big\}\Big\}^{\mathrm{T}} \tag{A.20b}$$

$$\text{Sigma-point weights} \qquad \lambda = \alpha^2(n_x + \kappa) - n_x\,,\ w_{\mathrm{p}} = \sqrt{n_x + \lambda} \tag{A.20c}$$

$$i = 0 \Rightarrow \qquad w_{\mathrm{m},0} = \frac{\lambda}{n_x + \lambda},\ w_{\mathrm{c},0} = \frac{\lambda}{n_x + \lambda} + (1 - \alpha^2 + \beta) \tag{A.20d}$$

$$i = 1 \ldots 2n_x \Rightarrow \qquad w_{\mathrm{m},i} = \frac{1}{2(n_x + \lambda)},\ w_{\mathrm{c},i} = \frac{1}{2(n_x + \lambda)} \tag{A.20e}$$

Prediction step:

$$\text{New sigma-points} \qquad \boldsymbol{\mathcal{X}}_{k-1}^+ = \widehat{\mathbf{X}}_{k-1}^+ + w_{\mathrm{p}}\left[0,\ \mathbf{S}_{k-1}^+, -\mathbf{S}_{k-1}^+\right] \tag{A.20f}$$

$$\text{Propagated sigma-points} \qquad \boldsymbol{\mathcal{X}}_k^* = \boldsymbol{f}(\boldsymbol{\mathcal{X}}_{k-1}^+, \boldsymbol{u}_{k-1}) \tag{A.20g}$$

$$\text{Predicted state estimate} \qquad \hat{\boldsymbol{x}}_k^- = \sum_{i=0}^{2n_x} w_{\mathrm{m},i}\, \boldsymbol{\mathcal{X}}_k^*(:,i) \tag{A.20h}$$

$$\text{Prelim. predicted cov. MSR} \qquad \mathbf{S}_k^- = \mathrm{qr}\Big\{\Big[\sqrt{w_{\mathrm{c},1}}\Big(\boldsymbol{\mathcal{X}}_k^*(:,1\!:\!2n_x) - \widehat{\mathbf{X}}_k^-\Big)\ ,\ \mathbf{S}_k^{\mathrm{Q}}\Big]\Big\}^{\mathrm{T}} \tag{A.20i}$$

$$\text{Predicted cov. MSR} \qquad \mathbf{S}_k^- = \mathrm{cholupdate}\Big\{\mathbf{S}_k^-, \boldsymbol{\mathcal{X}}_k^*(:,0) - \hat{\boldsymbol{x}}_k^-, w_{\mathrm{c},0}\Big\} \tag{A.20j}$$

Correction step:

$$\text{New sigma-points} \qquad \boldsymbol{\mathcal{X}}_k^- = \widehat{\mathbf{X}}_k^- + w_{\mathrm{p}}\left[0,\ \mathbf{S}_k^-, -\mathbf{S}_k^-\right] \tag{A.20k}$$

$$\text{Propagated sigma-points} \qquad \boldsymbol{\mathcal{Y}}_k^* = \boldsymbol{h}(\boldsymbol{\mathcal{X}}_k^-, \boldsymbol{u}_k) \tag{A.20l}$$

$$\text{Predicted output} \qquad \hat{\boldsymbol{y}}_k = \sum_{i=0}^{2n_x} w_{\mathrm{m},i}\, \boldsymbol{\mathcal{Y}}_k^*(:,i) \tag{A.20m}$$

$$\text{Cross covariance matrix} \qquad \mathbf{P}_k^{xy} = \sum_{i=0}^{2n_x} w_{\mathrm{c},i}\Big(\boldsymbol{\mathcal{X}}_k^*(:,i) - \widehat{\mathbf{X}}_k^-\Big)\Big(\boldsymbol{\mathcal{Y}}_k^*(:,i) - \widehat{\mathbf{Y}}_k\Big)^{\mathrm{T}} \tag{A.20n}$$

$$\text{Prelim. Innovation cov. MSR} \qquad \mathbf{S}_k^{yy} = \mathrm{qr}\Big\{\Big[\sqrt{w_{\mathrm{c},1}}\Big(\boldsymbol{\mathcal{Y}}_k^*(:,1\!:\!2n_x) - \widehat{\mathbf{Y}}_k\Big)\ ,\ \mathbf{S}_k^{\mathrm{R}}\Big]\Big\}^{\mathrm{T}} \tag{A.20o}$$

$$\text{Innovation cov. MSR} \qquad \mathbf{S}_k^{yy} = \mathrm{cholupdate}\{\mathbf{S}_k^{yy}, \boldsymbol{\mathcal{Y}}_k^*(:,0) - \hat{\boldsymbol{y}}_k, w_{\mathrm{c},0}\} \tag{A.20p}$$

$$\text{Kalman Gain} \qquad \boldsymbol{\mathcal{K}}_k = \Big(\mathbf{P}_k^{xy} / (\mathbf{S}_k^{yy})^{\mathrm{T}}\Big) / \mathbf{S}_k^{yy} \tag{A.20q}$$

$$\text{Corrected covariance MSR} \qquad \mathbf{S}_k^+ = \mathrm{cholupdate}\Big\{\mathbf{S}_k^-, \boldsymbol{\mathcal{K}}_k \mathbf{S}_k^{yy}, -1\Big\} \tag{A.20r}$$

$$\text{Filter Innovation} \qquad \boldsymbol{v}_k = \boldsymbol{y}_k - \hat{\boldsymbol{y}}_k^- \tag{A.20s}$$

$$\text{Corrected state estimate} \qquad \hat{\boldsymbol{x}}_k^+ = \hat{\boldsymbol{x}}_k^- + \boldsymbol{\mathcal{K}}_k \boldsymbol{v}_k \tag{A.20t}$$

A.4 The Slave Cubature Kalman Filter

In principle, the Algorithm 2.3 (p. 24) for the CKF can be used also as a nonlinear slave filter. This leads to a slightly different realization which is summed up in Algo. A.3.

Algorithm A.3 (The Slave Cubature Kalman Filter)

Initialization step:

$$\text{Initial state mean} \qquad \hat{\boldsymbol{x}}^{+}_{\mathrm{s},0} = E\{\boldsymbol{x}_{\mathrm{s},0}\} \tag{A.21a}$$

$$\text{Initial covariance} \qquad \mathbf{P}^{+}_{\mathrm{s},0} = E\Big\{(\boldsymbol{x}_{\mathrm{s},0} - \hat{\boldsymbol{x}}_{\mathrm{s},0})(\boldsymbol{x}_{\mathrm{s},0} - \hat{\boldsymbol{x}}_{\mathrm{s},0})^{\mathrm{T}}\Big\} \tag{A.21b}$$

$$\text{Sigma-point weights} \qquad w_{\mathrm{m}} = (2n_x)^{-1}\,,\; w_{\mathrm{p}} = \sqrt{n_x} \tag{A.21c}$$

Prediction step:

$$\text{New sigma-points} \qquad \boldsymbol{\mathcal{X}}^{+}_{\mathrm{s},k-1} = \widehat{\mathbf{X}}^{+}_{\mathrm{s},k-1} + w_{\mathrm{p}}\Big[\sqrt{\mathbf{P}^{+}_{\mathrm{s},k-1}}, -\sqrt{\mathbf{P}^{+}_{\mathrm{s},k-1}}\Big] \tag{A.21d}$$

$$\text{Propagated sigma-points} \qquad \boldsymbol{\mathcal{X}}^{*}_{\mathrm{s},k} = \boldsymbol{f}_{\mathrm{s}}(\boldsymbol{\mathcal{X}}^{+}_{\mathrm{s},k-1}) \tag{A.21e}$$

$$\text{Predicted state estimate} \qquad \hat{\boldsymbol{x}}^{-}_{\mathrm{s},k} = w_{\mathrm{m}} \sum_{i=1}^{2n_x} \boldsymbol{\mathcal{X}}^{*}_{\mathrm{s},k}(:,i) \tag{A.21f}$$

$$\text{Predicted covariance matrix} \qquad \mathbf{P}^{-}_{\mathrm{s},k} = w_{\mathrm{m}} \boldsymbol{\mathcal{X}}^{*}_{\mathrm{s},k}\Big(\boldsymbol{\mathcal{X}}^{*}_{\mathrm{s},k}\Big)^{\mathrm{T}} - \hat{\boldsymbol{x}}^{-}_{\mathrm{s},k}\Big(\hat{\boldsymbol{x}}^{-}_{\mathrm{s},k}\Big)^{\mathrm{T}} + \mathbf{Q}_{\mathrm{s},k-1} \tag{A.21g}$$

Correction step:

$$\text{New sigma-points} \qquad \boldsymbol{\mathcal{X}}^{-}_{\mathrm{s},k} = \widehat{\mathbf{X}}^{-}_{\mathrm{s},k} + w_{\mathrm{p}}\Big[\sqrt{\mathbf{P}^{-}_{\mathrm{s},k}}, -\sqrt{\mathbf{P}^{-}_{\mathrm{s},k}}\Big] \tag{A.21h}$$

$$\text{Propagated sigma-points} \qquad \boldsymbol{\mathcal{Y}}^{*}_{\mathrm{s},k} = \boldsymbol{h}_{\mathrm{s}}(\boldsymbol{\mathcal{X}}^{-}_{\mathrm{s},k}, \boldsymbol{u}_{\mathrm{s},k}) \tag{A.21i}$$

$$\text{Predicted output} \qquad \hat{\boldsymbol{y}}_{\mathrm{s},k} = w_{\mathrm{m}} \sum_{i=1}^{2n_x} \boldsymbol{\mathcal{Y}}^{*}_{\mathrm{s},k}(:,i) \tag{A.21j}$$

$$\text{Cross covariance matrix} \qquad \mathbf{P}^{xy}_{\mathrm{s},k} = w_{\mathrm{m}}\, \boldsymbol{\mathcal{X}}^{*}_{\mathrm{s},k}\left(\boldsymbol{\mathcal{Y}}^{*}_{\mathrm{s},k}\right)^{\mathrm{T}} - \hat{\boldsymbol{x}}^{-}_{\mathrm{s},k}\Big(\hat{\boldsymbol{y}}_{\mathrm{s},k}\Big)^{\mathrm{T}} \tag{A.21k}$$

$$\text{Innovation covariance matrix} \qquad \mathbf{P}^{yy}_{\mathrm{s},k} = w_{\mathrm{m}}\, \boldsymbol{\mathcal{Y}}^{*}_{\mathrm{s},k}\left(\boldsymbol{\mathcal{Y}}^{*}_{\mathrm{s},k}\right)^{\mathrm{T}} - \hat{\boldsymbol{y}}_{\mathrm{s},k}\Big(\hat{\boldsymbol{y}}_{\mathrm{s},k}\Big)^{\mathrm{T}} + \mathbf{R}_{\mathrm{s},k} \tag{A.21l}$$

$$\text{Kalman Gain} \qquad \boldsymbol{\mathcal{K}}_{\mathrm{s},k} = \mathbf{P}^{xy}_{\mathrm{s},k}(\mathbf{P}^{yy}_{\mathrm{s},k})^{-1} \tag{A.21m}$$

$$\text{Corrected covariance matrix} \qquad \mathbf{P}^{+}_{\mathrm{s},k} = \mathbf{P}^{-}_{\mathrm{s},k} - \boldsymbol{\mathcal{K}}_{\mathrm{s},k}\mathbf{P}^{yy}_{\mathrm{s},k}\boldsymbol{\mathcal{K}}^{\mathrm{T}}_{\mathrm{s},k} \tag{A.21n}$$

$$\text{Filter Innovation} \qquad \boldsymbol{v}_{\mathrm{s},k} = \boldsymbol{y}_{\mathrm{s},k} - \hat{\boldsymbol{y}}_{\mathrm{s},k} \tag{A.21o}$$

$$\text{Corrected state estimate} \qquad \hat{\boldsymbol{x}}^{+}_{\mathrm{s},k} = \hat{\boldsymbol{x}}^{-}_{\mathrm{s},k} + \boldsymbol{\mathcal{K}}_{\mathrm{s},k}\boldsymbol{v}_{\mathrm{s},k} \tag{A.21p}$$

Therein, $\boldsymbol{f}_{\mathrm{s}}(\cdot)$ is the process model for the noise states and $\boldsymbol{h}_{\mathrm{s}}(\cdot)$ is measurement model. The *control inputs* $\boldsymbol{u}_{\mathrm{s},k}$ and measurements $\boldsymbol{y}_{\mathrm{s},k}$ are defined according to Eq. (2.47a) - (2.47k) (cf. p. 39). Depending on the selected noise variables (process or measurement noise) and the selected *measurements* (innovation or state residual), the inputs and outputs look differently.

For instance, the measurement noise estimation (as discussed in Sect. 5.4, p. 113) is conducted with the inputs $\boldsymbol{u}_{\mathrm{s},k} = \mathrm{diag}\Big\{\mathbf{P}^{yy}_{k} - \hat{\mathbf{R}}_{k-1}\Big\}$ and the measurements $\boldsymbol{y}_{\mathrm{s},k} = \mathrm{diag}\Big\{\hat{\mathbf{P}}^{yy}_{k}\Big\}$.

A.5 Derivation of the Linear Slave Kalman Filter

This section provides some derivations on the linear slave KF that has been introduced previously in Algorithm 2.5. In particular, the output matrix $\mathbf{C}_{\mathrm{s},k}$, the control input $\boldsymbol{u}_{\mathrm{s},k}$ and the observed output $\hat{\boldsymbol{y}}_{\mathrm{s},k}$ of the slave filter are derived for different options, for instance, using the Eqs. (2.16g), (2.16l) and (2.16n) of the CKF. These are generalized as

$$\mathbf{P}_k^- = \mathbf{H}_k^{xx} + \mathbf{Q}_k \tag{A.22}$$

$$\mathbf{P}_k^{yy} = \mathbf{H}_k^{yy} + \mathbf{R}_k \tag{A.23}$$

$$\widetilde{\mathbf{P}}_k = \boldsymbol{\mathcal{K}}_k \mathbf{P}_k^{yy} \boldsymbol{\mathcal{K}}_k^{\mathrm{T}} = \mathbf{P}_k^- - \mathbf{P}_k^+ \tag{A.24}$$

which are applicable for linear and nonlinear Kalman filters. Therein, only the matrices $\mathbf{H}_k^{xx}$ and $\mathbf{H}_k^{yy}$ are filter-specific quantities, e.g. $\mathbf{H}_k^{xx} = \mathbf{A}_k \mathbf{P}_{k-1}^+ \mathbf{A}_k^{\mathrm{T}}$ and $\mathbf{H}_k^{yy} = \mathbf{C}_k \mathbf{P}_k^- \mathbf{C}_k^{\mathrm{T}}$ for the linear time-variant KF (cf. Algo. 2.1, p. 20).

A.5.1 The Innovation-Based Approach

The innovation-based approach

$$\boldsymbol{v}_{\mathrm{s},k} = \boldsymbol{y}_{\mathrm{s},k} - \hat{\boldsymbol{y}}_{\mathrm{s},k} = \mathrm{diag}\Big\{\hat{\mathbf{P}}_k^{yy}\Big\} - \mathrm{diag}\Big\{\mathbf{P}_k^{yy}\Big\} \tag{A.25}$$

uses the diagonal elements of the innovation covariances matrix as measurements (or observations respectively) of the slave filter. The measurement $\boldsymbol{y}_{\mathrm{s},k}$ is defined by Eq. (2.27). The observation model $\hat{\boldsymbol{y}}_{\mathrm{s},k}$ for process noise estimation is derived as follows:

$$\begin{aligned}
\mathrm{diag}\Big\{\mathbf{P}_k^{yy}\Big\} &= \mathrm{diag}\Big\{\boldsymbol{\mathcal{K}}_k^{\#} \widetilde{\mathbf{P}}_k \Big(\boldsymbol{\mathcal{K}}_k^{\#}\Big)^{\mathrm{T}}\Big\} && \text{(A.26)}\\
&= \mathrm{diag}\Big\{\boldsymbol{\mathcal{K}}_k^{\#} \Big(\mathbf{P}_k^- - \mathbf{P}_k^+\Big) \Big(\boldsymbol{\mathcal{K}}_k^{\#}\Big)^{\mathrm{T}}\Big\} && \text{(A.27)}\\
&= \mathrm{diag}\Big\{\boldsymbol{\mathcal{K}}_k^{\#} \Big(\mathbf{H}_k^{xx} + \hat{\mathbf{Q}}_k - \mathbf{P}_k^+\Big) \Big(\boldsymbol{\mathcal{K}}_k^{\#}\Big)^{\mathrm{T}}\Big\} && \text{(A.28)}\\
&= \mathrm{diag}\Big\{\boldsymbol{\mathcal{K}}_k^{\#} \hat{\mathbf{Q}}_k \Big(\boldsymbol{\mathcal{K}}_k^{\#}\Big)^{\mathrm{T}}\Big\} + \mathrm{diag}\Big\{\boldsymbol{\mathcal{K}}_k^{\#} \Big(\mathbf{H}_k^{xx} - \mathbf{P}_k^+\Big) \Big(\boldsymbol{\mathcal{K}}_k^{\#}\Big)^{\mathrm{T}}\Big\} && \text{(A.29)}\\
&= \underbrace{\Big(\boldsymbol{\mathcal{K}}_k^{\#} \circ \boldsymbol{\mathcal{K}}_k^{\#}\Big)}_{\mathbf{C}_{\mathrm{s},k}} \underbrace{\mathrm{diag}\Big\{\hat{\mathbf{Q}}_k\Big\}}_{\hat{\boldsymbol{x}}_{\mathrm{s},k}} + \underbrace{\mathrm{diag}\Big\{\boldsymbol{\mathcal{K}}_k^{\#} \Big(\mathbf{H}_k^{xx} - \mathbf{P}_k^+\Big) \Big(\boldsymbol{\mathcal{K}}_k^{\#}\Big)^{\mathrm{T}}\Big\}}_{\boldsymbol{u}_{\mathrm{s},k}} && \text{(A.30)}
\end{aligned}$$

Note that, the true covariance matrix $\mathbf{Q}_k$ in Eq. (A.28) has been replaced by its estimate $\hat{\mathbf{Q}}_k$. Additionally, the following relationship

$$\mathbf{H}_k^{xx} - \mathbf{P}_k^+ = \mathbf{P}_k^- - \hat{\mathbf{Q}}_{k-1} - \mathbf{P}_k^+ = \widetilde{\mathbf{P}}_k - \hat{\mathbf{Q}}_{k-1}\ . \tag{A.31}$$

can be used instead. This shows directly the recursive nature of the slave filter and avoids also another evaluation of sigma-points. $\hat{\mathbf{Q}}_{k-1}$ is the previous process noise estimate used by the master filter in time step t_k. A disadvantage of this approach is that the pseudo-inverse $\boldsymbol{\mathcal{K}}_k^{\#}$ needs to be evaluated in every recursion step which though is avoidable using the residual-based approach (Sect. A.5.2).

The observation model for measurement noise estimation simply denotes:

$$\operatorname{diag}\{\mathbf{P}_k^{yy}\} = \operatorname{diag}\left\{\mathbf{H}_k^{yy} + \hat{\mathbf{R}}_k\right\} \tag{A.32}$$

$$= \underbrace{\mathbf{I}}_{\mathbf{C}_{\mathrm{s},k}} \underbrace{\operatorname{diag}\left\{\hat{\mathbf{R}}_k\right\}}_{\hat{\boldsymbol{x}}_{\mathrm{s},k}} + \underbrace{\operatorname{diag}\{\mathbf{H}_k^{yy}\}}_{\boldsymbol{u}_{\mathrm{s},k}} \tag{A.33}$$

Note that again, the covariance matrix $\mathbf{R}_k$ has been replaced by its respective estimate $\hat{\mathbf{R}}_k$. Moreover, the matrix $\mathbf{H}_k^{yy}$ is approximated by

$$\mathbf{H}_k^{yy} \approx \mathbf{P}_k^{yy} - \hat{\mathbf{R}}_{k-1} \ . \tag{A.34}$$

A.5.2 The Residual-Based Approach

Contrary, the residual-based approach does not use the innovation but the slave filter's state residual

$$\widetilde{\boldsymbol{x}}_{\mathrm{s},k} = \hat{\boldsymbol{x}}_{\mathrm{s},k}^{+} - \hat{\boldsymbol{x}}_{\mathrm{s},k}^{-} = \operatorname{diag}\left\{\hat{\widetilde{\mathbf{P}}}_k\right\} - \operatorname{diag}\left\{\widetilde{\mathbf{P}}_k\right\} \tag{A.35}$$

to estimate the process or measurement noise. For process noise estimation the relationship constitutes as follows:

$$\operatorname{diag}\left\{\widetilde{\mathbf{P}}_k\right\} = \operatorname{diag}\left\{\mathbf{P}_k^{-} - \mathbf{P}_k^{+}\right\} \tag{A.36}$$

$$= \operatorname{diag}\left\{\mathbf{H}_k^{xx} + \hat{\mathbf{Q}}_k - \mathbf{P}_k^{+}\right\} \tag{A.37}$$

$$= \underbrace{\mathbf{I}}_{\mathbf{C}_{\mathrm{s},k}} \underbrace{\operatorname{diag}\left\{\hat{\mathbf{Q}}_k\right\}}_{\hat{\boldsymbol{x}}_{\mathrm{s},k}} + \underbrace{\operatorname{diag}\left\{\mathbf{H}_k^{xx} - \mathbf{P}_k^{+}\right\}}_{\boldsymbol{u}_{\mathrm{s},k}} \tag{A.38}$$

For measurement noise estimation the relationship is:

$$\operatorname{diag}\left\{\widetilde{\mathbf{P}}_k\right\} = \operatorname{diag}\left\{\mathbf{P}_k^{-} - \mathbf{P}_k^{+}\right\} \tag{A.39}$$

$$= \operatorname{diag}\left\{\boldsymbol{\mathcal{K}}_k \mathbf{P}_k^{yy} \boldsymbol{\mathcal{K}}_k^{\mathrm{T}}\right\} \tag{A.40}$$

$$= \operatorname{diag}\left\{\boldsymbol{\mathcal{K}}_k \left(\mathbf{H}_k^{yy} + \hat{\mathbf{R}}_k\right) \boldsymbol{\mathcal{K}}_k^{\mathrm{T}}\right\} \tag{A.41}$$

$$= \underbrace{\left(\boldsymbol{\mathcal{K}}_k \circ \boldsymbol{\mathcal{K}}_k^{\mathrm{T}}\right)}_{\mathbf{C}_{\mathrm{s},k}} \underbrace{\operatorname{diag}\left\{\hat{\mathbf{R}}_k\right\}}_{\hat{\boldsymbol{x}}_{\mathrm{s},k}} + \underbrace{\operatorname{diag}\left\{\boldsymbol{\mathcal{K}}_k \mathbf{H}_k^{yy} \boldsymbol{\mathcal{K}}_k^{\mathrm{T}}\right\}}_{\boldsymbol{u}_{\mathrm{s},k}} \tag{A.42}$$

Again, $\mathbf{H}_k^{xx}$ and $\mathbf{H}_k^{yy}$ can be replaced by Eqs. (A.31) and (A.34).

B Simulation Results

This appendix is intended to provide additional simulation results from simulation study described in Chapter 5. The AWS scenario is addressed in Figs. B.1, B.2 and B.3, showing the state estimation errors and the noise estimates. The TI scenario is shown in Figs. B.4 and B.5. The acid test scenarios for $v_\mathrm{m} = 9\,\mathrm{m/s}$ and $v_\mathrm{m} = 15\,\mathrm{m/s}$ are displayed comparatively in Fig. B.6. The corresponding state estimation results from the performance test are shown in Fig. B.7 and B.8 for 9 m/s and Fig. B.9 and B.10 for 15 m/s wind speed. The corresponding performance variables are displayed in Fig. B.11 and B.12 (p. 155) as well as Fig. B.13 and B.14 (p. 156).

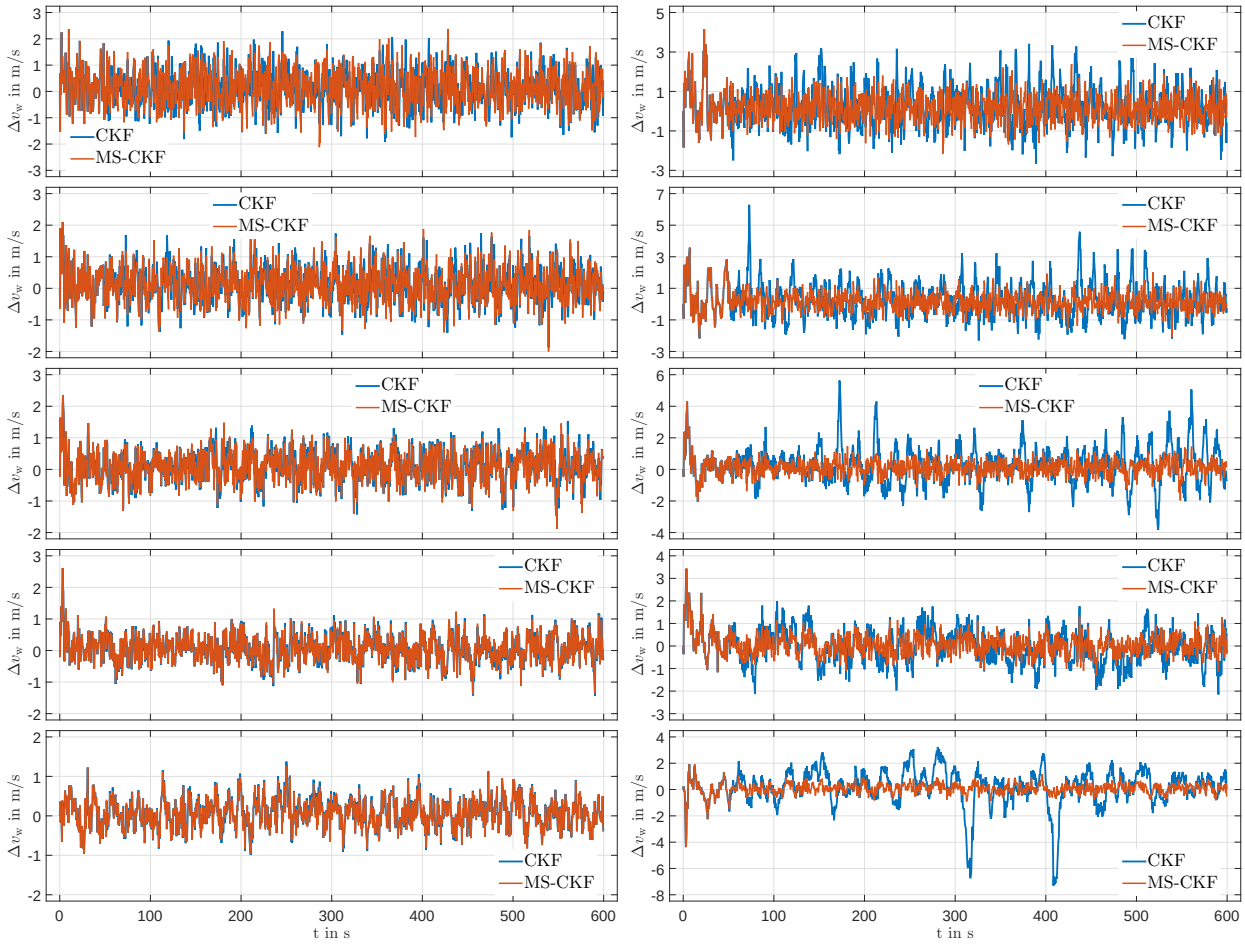

Figure B.1: Estimation results from the AWS scenario $v_\mathrm{m} \in [6, 9, 12, 15, 18]\,\mathrm{m/s}$ (from the bottom up) with the low sensor noise environment comparing the wind speed estimation errors (left: designed for low noise level, right: designed for zero noise), cf. Fig. 5.5 on p. 114

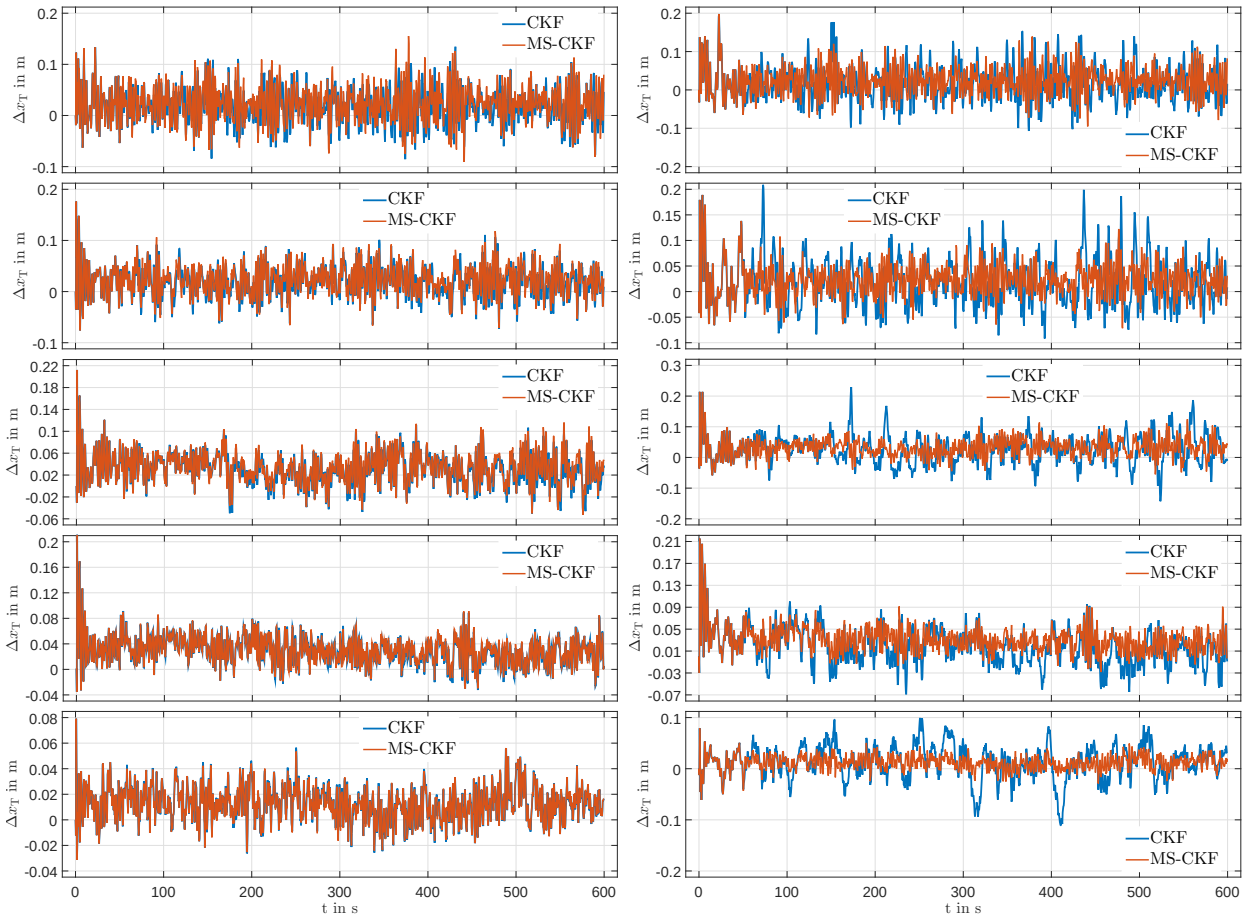

Figure B.2: Estimation results from the AWS scenario $v_\mathrm{m} \in [6, 9, 12, 15, 18]\,\mathrm{m/s}$ (from the bottom up) with the low sensor noise environment comparing the nacelle position estimates (left: designed for low noise level, right: designed for zero noise), cf. Fig. 5.6 on p. 115

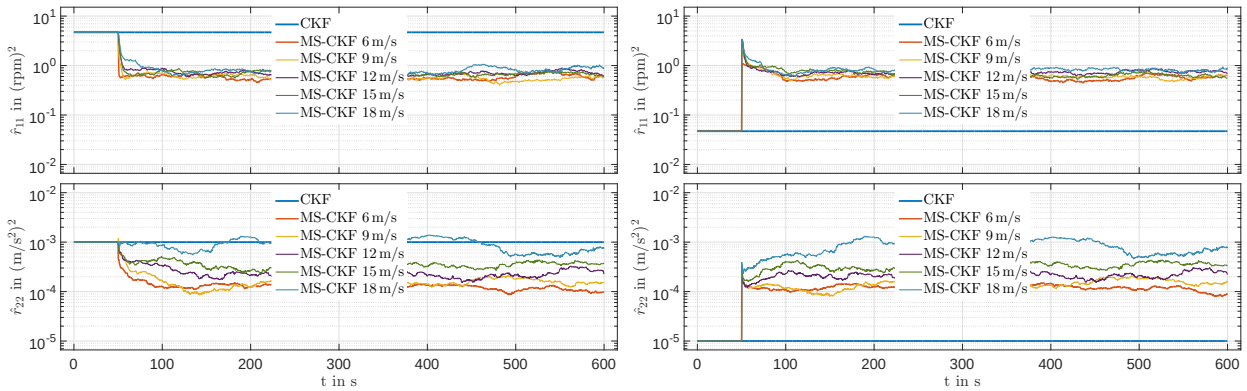

Figure B.3: Estimation results from the AWS scenario with low sensor noise (left: designed for medium noise level, right: designed for zero noise), cf. Fig. 5.7 on p. 116

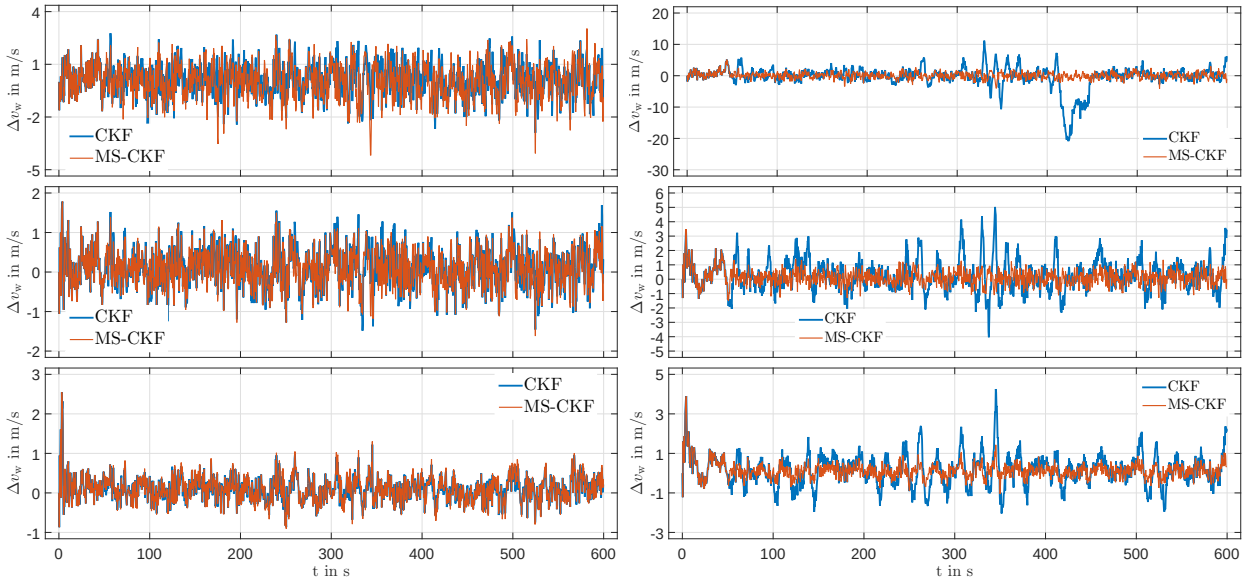

Figure B.4: Estimation results from the TI scenario with low sensor noise comparing wind speed estimates (left: designed for low noise level, right: designed for zero noise) for three different turbulence intensities from low TI (bottom) to high TI (top) with $v_\mathrm{m} = 12\,\mathrm{m/s}$ (cf. Fig. 5.8, p. 117)

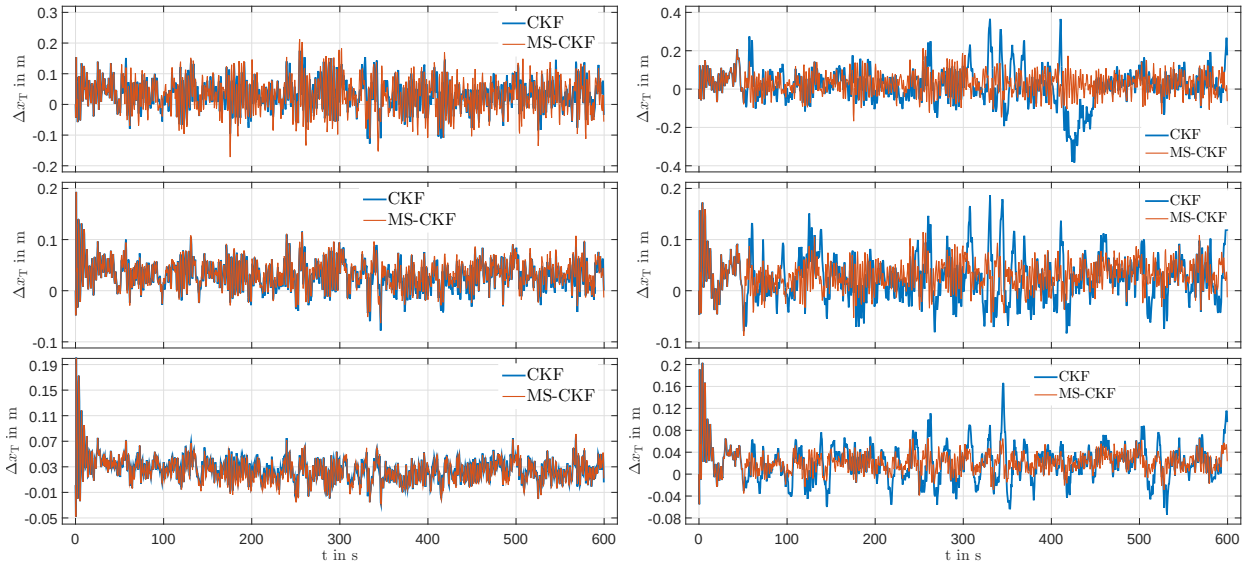

Figure B.5: Estimation results from the TI scenario with low sensor noise comparing nacelle position estimates (left: designed for low noise level, right: designed for zero noise) for three different turbulence intensities from low TI (bottom) to high TI (top) with $v_\mathrm{m} = 12\,\mathrm{m/s}$ (cf. Fig. 5.9, p. 118)

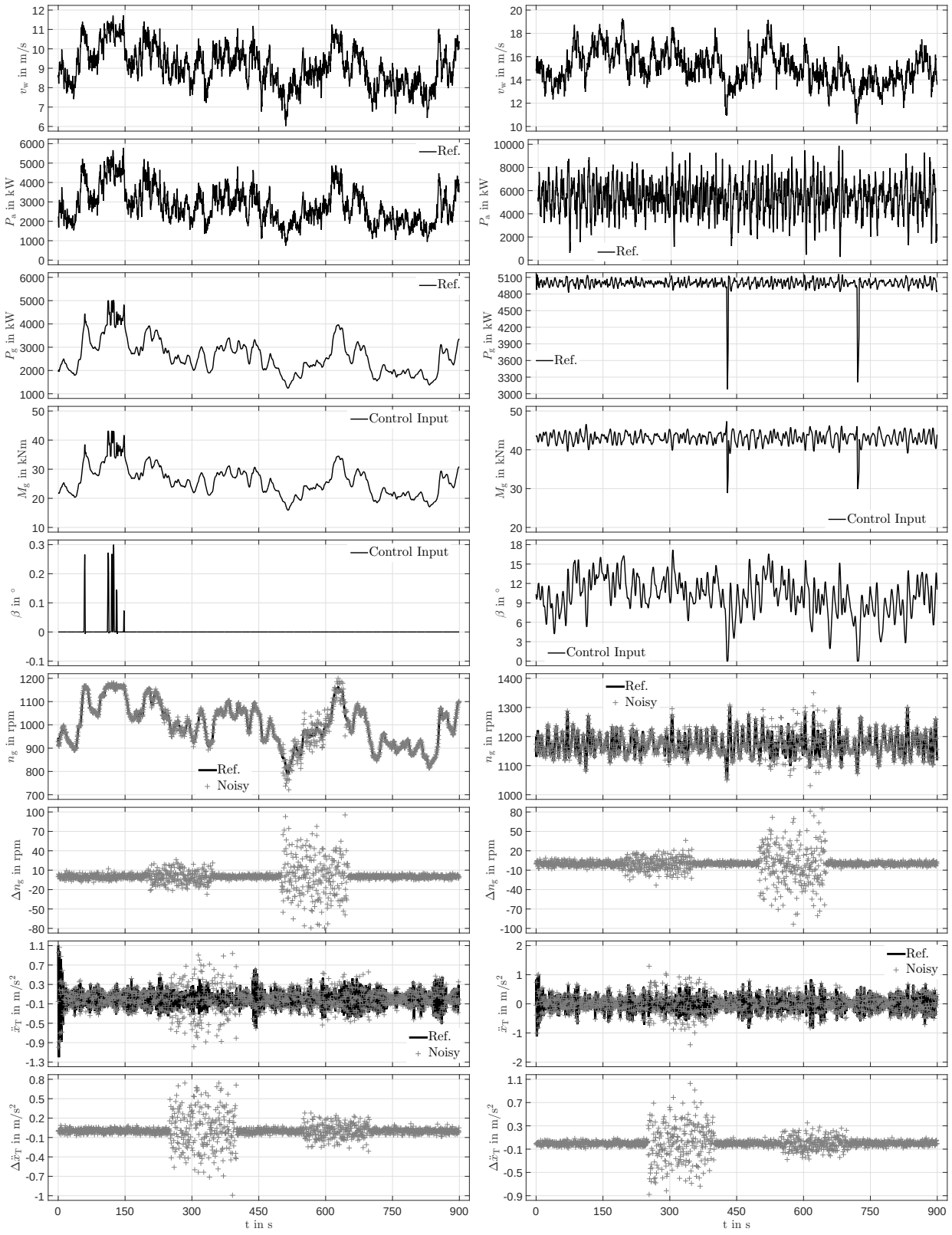

Figure B.6: Acid test scenarios with time-varying noise for $v_m = 9\,\mathrm{m/s}$ (left) and $v_m = 15\,\mathrm{m/s}$ (right) both with TI = 0.1 and $\kappa = 0.2$, cf. Fig. 5.4 on p. 113

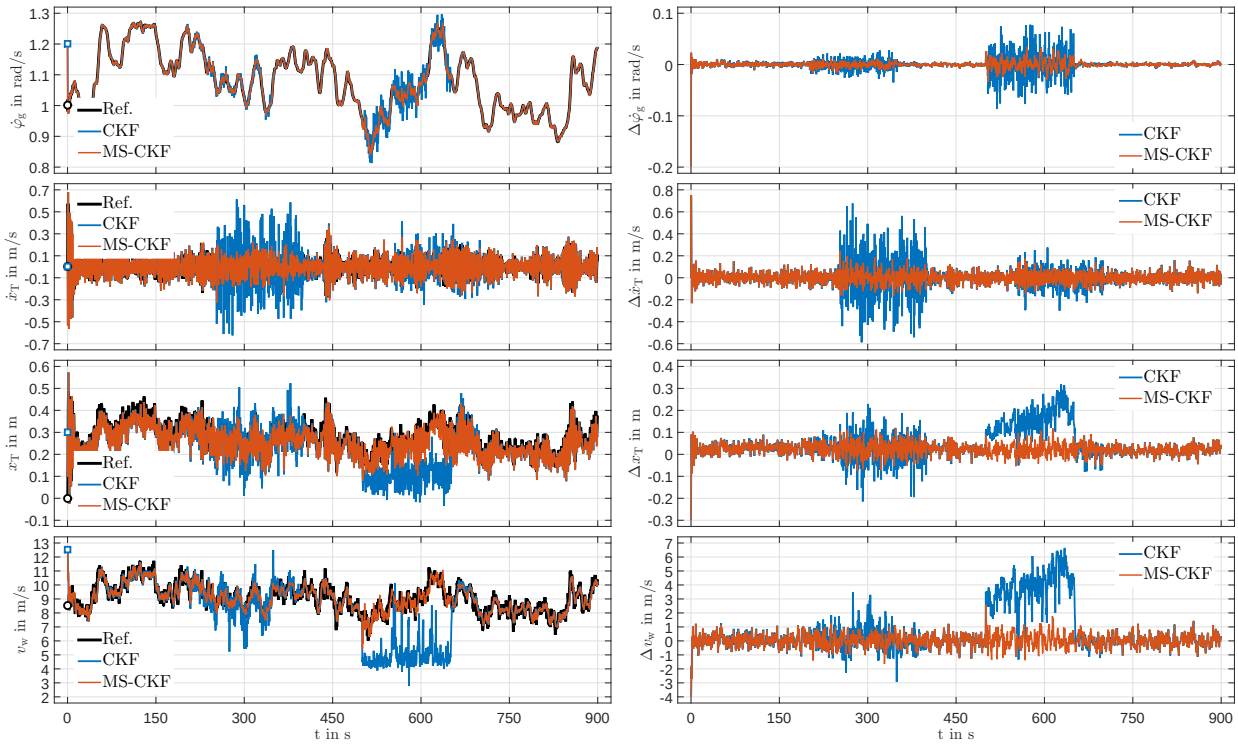

Figure B.7: Estimation results from the acid test scenario with $v_\mathrm{m} = 9\,\mathrm{m/s}$ and $\mathrm{TI} = 0.1$ as well as time-varying sensor noise (left: state estimates, right: estimation error), cf. Fig. 5.11 on p. 119

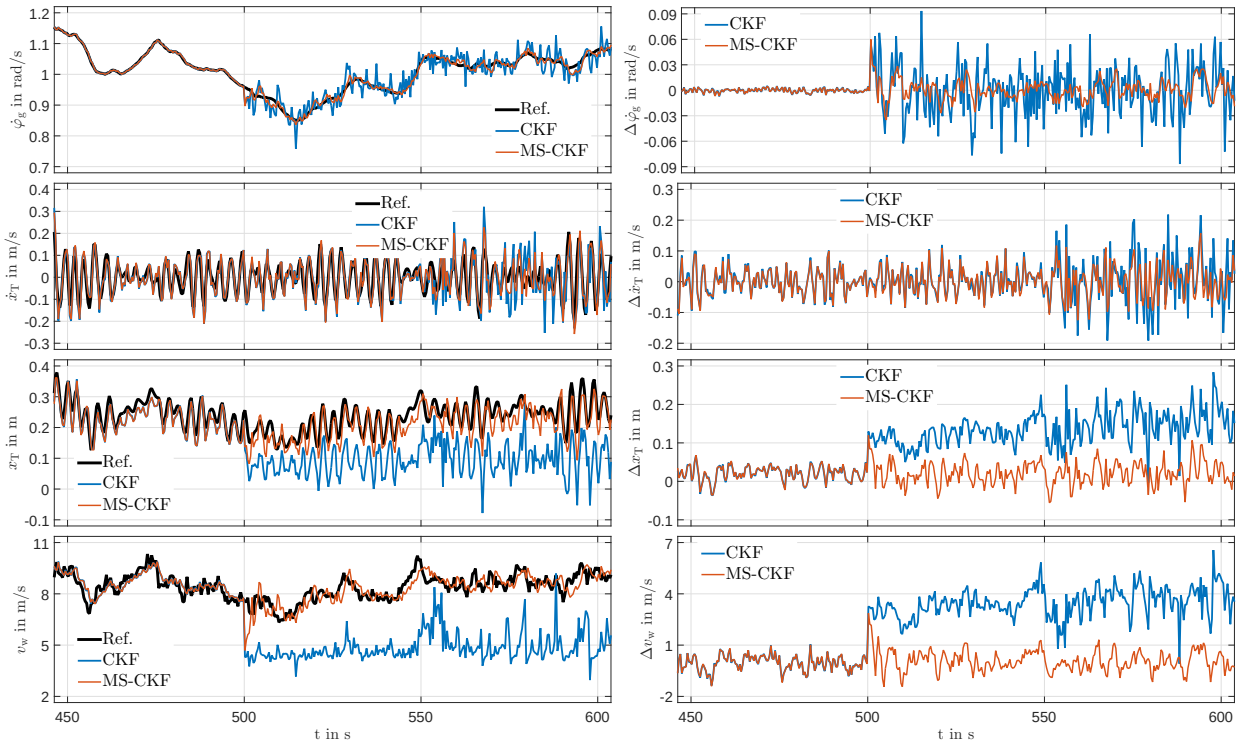

Figure B.8: Estimation results from the acid test scenario with $v_\mathrm{m} = 9\,\mathrm{m/s}$ and $\mathrm{TI} = 0.1$ as well as time-varying sensor noise (left: state estimates, right: estimation error), cf. Fig. 5.12 on p. 119

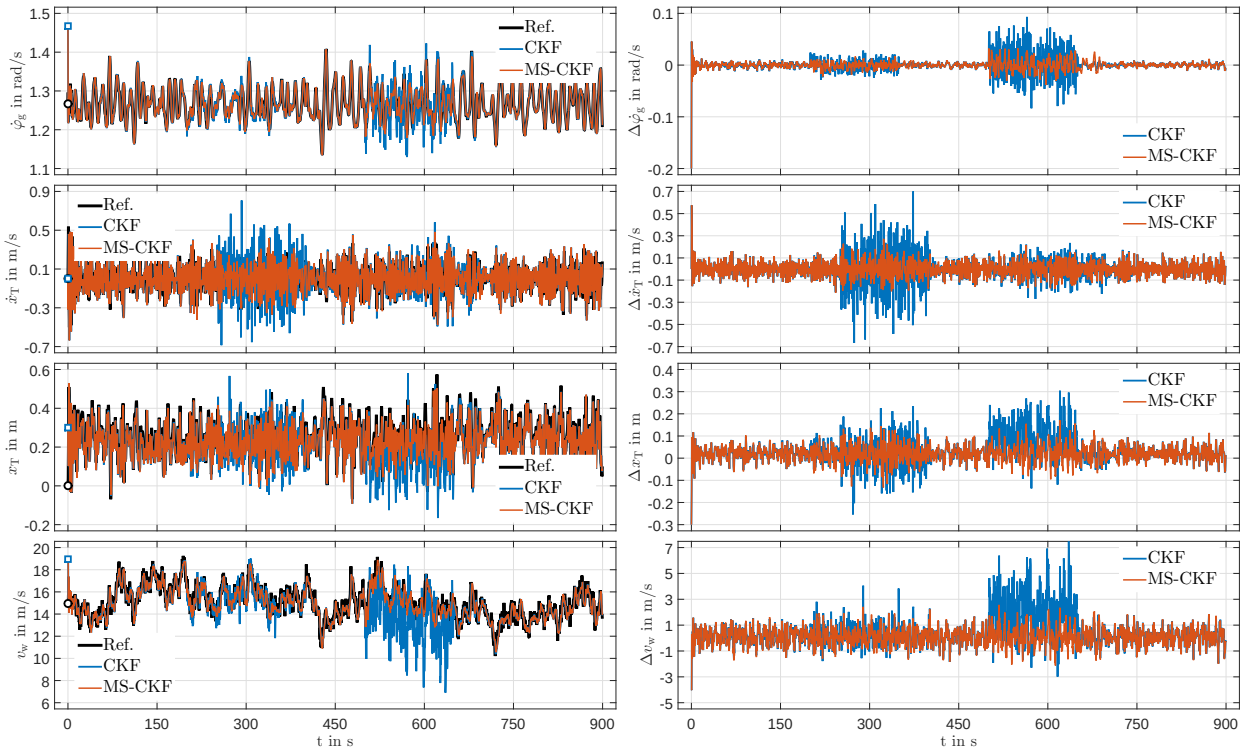

Figure B.9: Estimation results from the acid test scenario with $v_m = 15\,\mathrm{m/s}$ and $\mathrm{TI} = 0.1$ as well as time-varying sensor noise (left: state estimates, right: estimation error), cf. Fig. 5.11 on p. 119

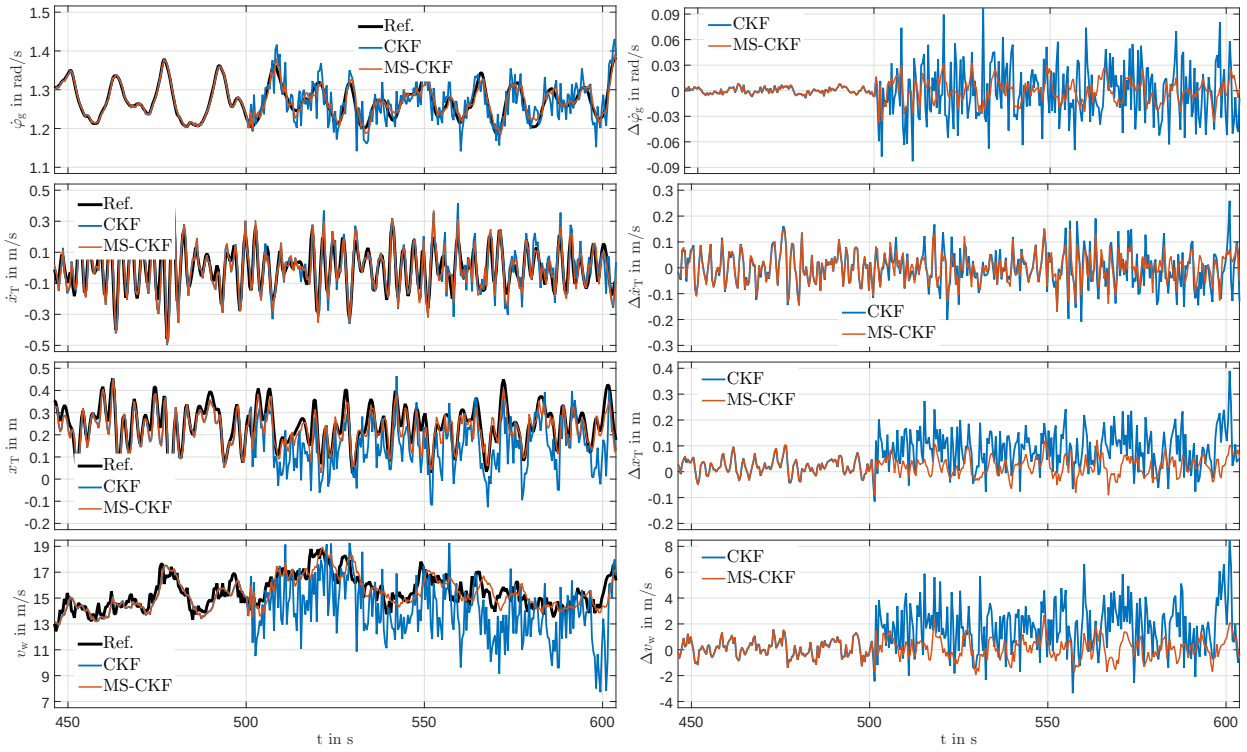

Figure B.10: Estimation results from the acid test scenario with $v_m = 15\,\mathrm{m/s}$ and $\mathrm{TI} = 0.1$ as well as time-varying sensor noise (left: state estimates, right: estimation error), cf. Fig. 5.12 on p. 119

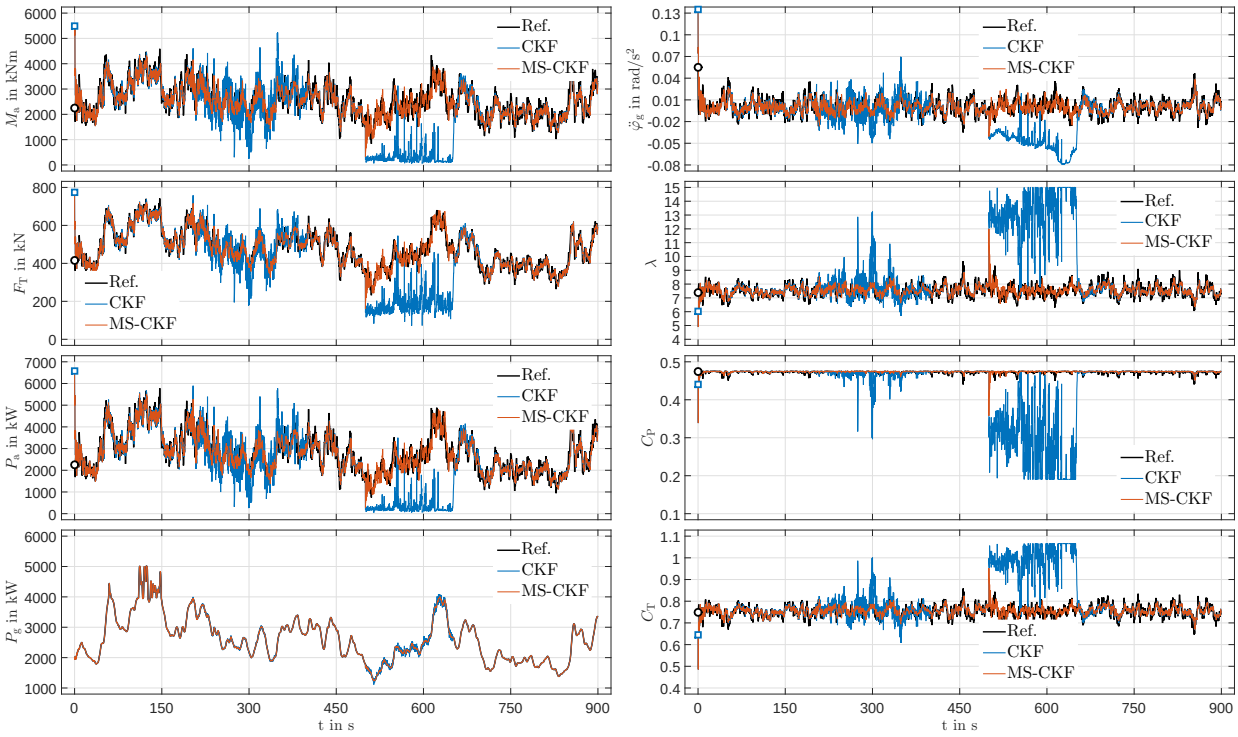

Figure B.11: Performance variables from the estimation results of the acid test scenario with $v_m = 9\,\mathrm{m/s}$ and TI = 0.1 as well as time-varying sensor noise (overview for $t \in [0\,,\,900]\mathrm{s}$), cf. Fig. 5.14 on p. 122

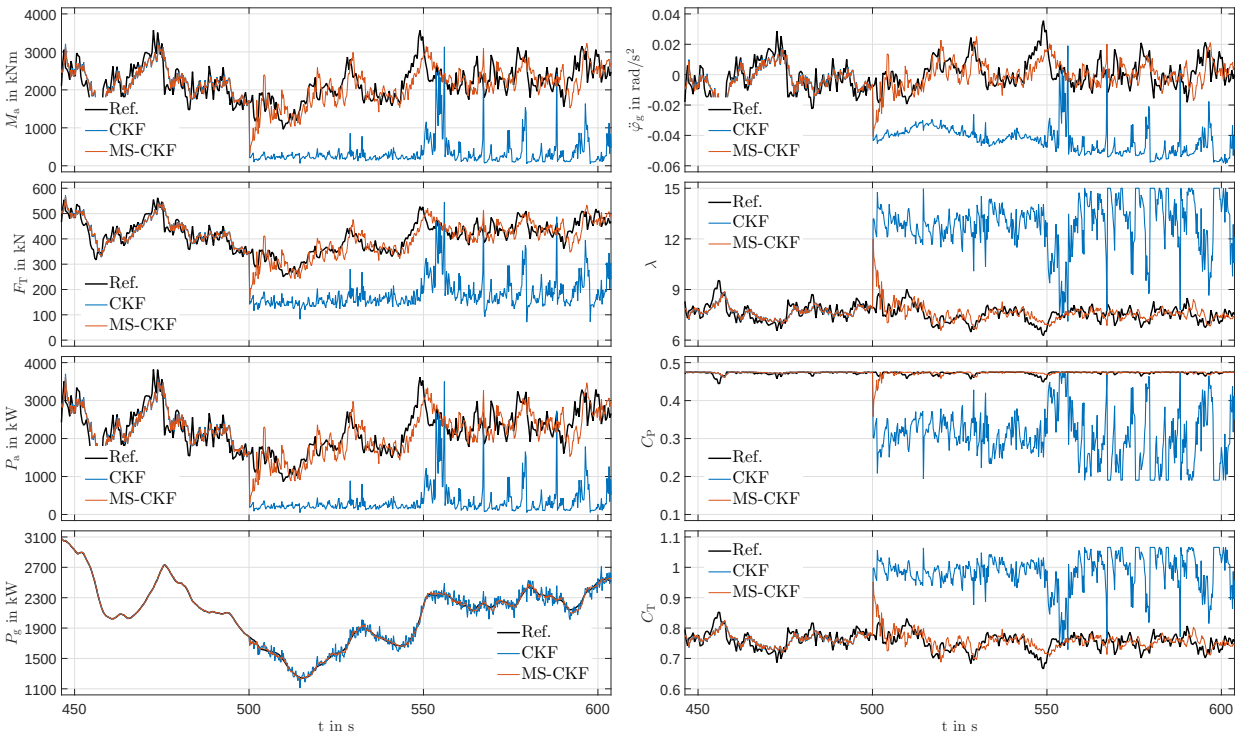

Figure B.12: Performance variables from the acid test scenario with $v_m = 9\,\mathrm{m/s}$ and TI = 0.1 as well as time-varying sensor noise (details for $t \in [450\,,\,600]\mathrm{s}$), cf. Fig. 5.15 on p. 122

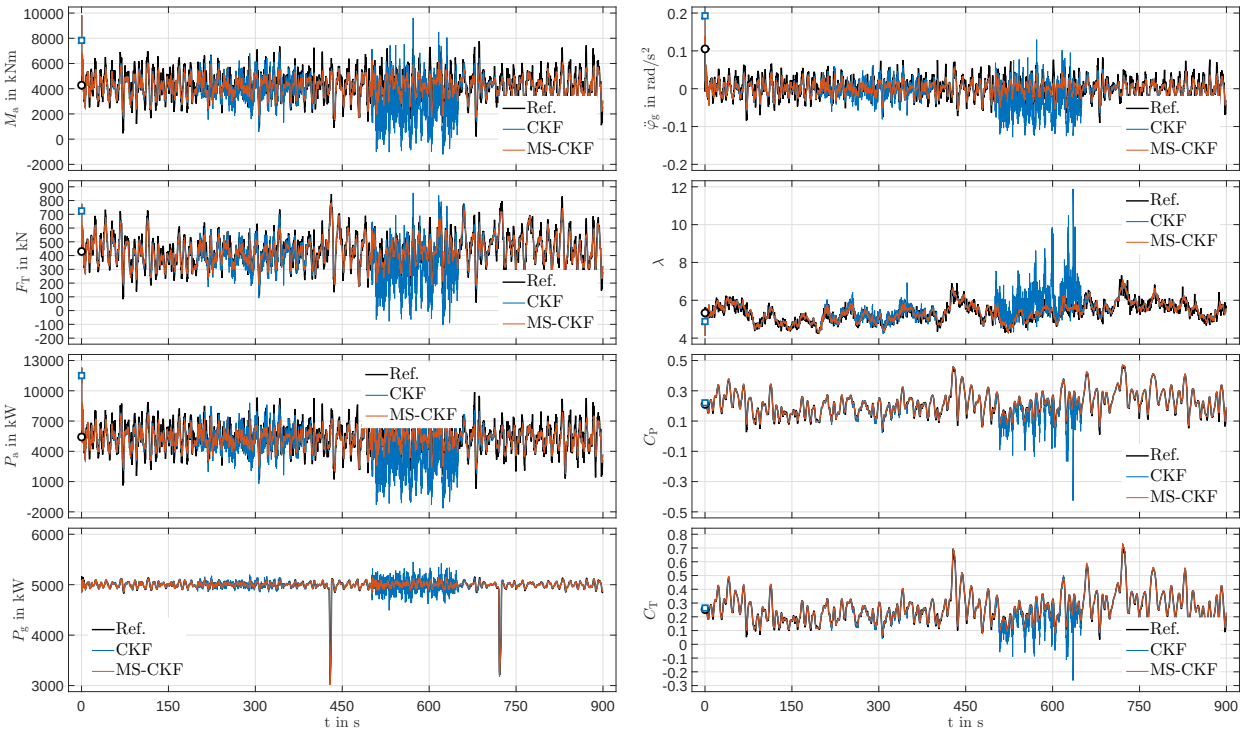

Figure B.13: Performance variables from the estimation results of the acid test scenario with $v_m = 15\,\mathrm{m/s}$ and TI = 0.1 as well as time-varying sensor noise (overview for $t \in [0\,,\,900]\mathrm{s}$), cf. Fig. 5.14 on p. 122

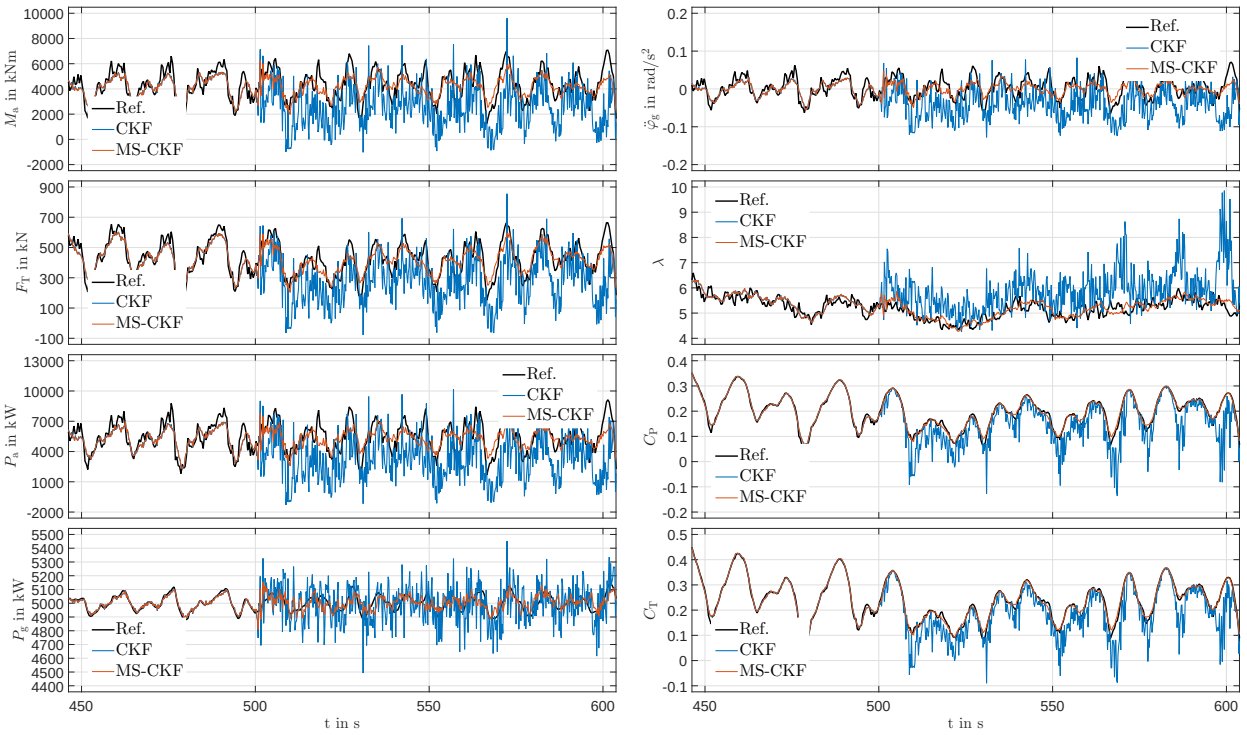

Figure B.14: Performance variables from the acid test scenario with $v_m = 15\,\mathrm{m/s}$ and TI = 0.1 as well as time-varying sensor noise (details for $t \in [450\,,\,600]\mathrm{s}$), cf. Fig. 5.15 on p. 122

C Own Publications

The following Tab. C.1 contains the author's accompanying contributions to the field of nonlinear state estimation and control of wind and tidal turbines.

Table C.1: Conference Contributions with Written Papers

	Reference Description	Bib.	Year
10	Thomas Schmitt, **Bastian Ritter**: Data-based identifiability and observability assessment for nonlinear control systems using the profile likelihood method, 21st IFAC World Congress in Berlin, Germany		2020
9	**Bastian Ritter**, Edwin Mora, Axel Schild, Bart M. Doekemeijer, Ulrich Konigorski: Adaptive master-slave cubature Kalman filters subject to state inequality constraints for wind turbine state estimation, 2019 American Control Conference (ACC), 10. - 12.7., Philadelphia, PA, USA	[171]	2019
8	**Bastian Ritter**, Christian Schmitz, Peter F. Pelz: Control-oriented modelling and state estimation of tidal turbines with pitch control, Proc. of the 20th International Seminar on Hydropower Plants, 14. - 16.11., Vienna, Austria	[165]	2018
7	**Bastian Ritter**, Edwin Mora, Axel Schild: Observability analysis for horizontal axis wind turbines using empirical gramian matrices, Journal of Physics: Conference Series, Vol. 1102 (2018) 012021, WindEurope conference 2018 within the Global Wind Summit, 25. – 28.9., Hamburg, Germany	[169]	2018
6	**Bastian Ritter**, Edwin Mora, Thorsten Schlicht, Axel Schild, Ulrich Konigorski: Adaptive sigma-point Kalman filtering for wind turbine state and process noise estimation, Journal of Physics: Conference Series, Vol. 1037 (2018) 032003, 7th Science of Making Torque from Wind, TORQUE 2018, Milan, Italy	[166]	2018
5	**Bastian Ritter**, Axel Schild, Ulrich Konigorski: Making nonlinear state estimation techniques ready for use in industrial wind turbine control systems, Proc. of the WindEurope Summit 2016, 27. - 29.9., Hamburg	[170]	2016
4	**Bastian Ritter**, Axel Schild, Matthias Feldt, Ulrich Konigorski: The design of nonlinear observers for wind turbine dynamic state and parameter estimation, Journal of Physics: Conference Series, Vol. 753 (2016) 052029, 6th Science of Making Torque from Wind, TORQUE 2016, Munich, Germany	[174]	2016
3	**Bastian Ritter**, Niko Chrysalidis, Ulrich Konigorski: Application of nonlinear Kalman filters to wind turbine state observation, Proc. of 12th EAWE PhD Seminar on Wind Energy in Europe 2016, Copenhagen, Denmark		2016
2	**Bastian Ritter**, Ulrich Konigorski: Advanced multivariable control design for modern multi-MW wind turbines, Proc. of the 11th EAWE PhD Seminar on Wind Energy in Europe 2015, 23. - 25.9., Stuttgart, Germany		2015
1	**Bastian Ritter**, Holger Fürst, Ulrich Konigorski, Mike Eichhorn: Multivariable model for simulation and control design of wind turbines, Proc. of the 12th German Wind Energy Conference (DEWEK 2015), Bremen, Germany	[172]	2015

Bibliography

[1] Fabiano Daher Adegas. "Structured, Gain-Scheduled Control of Wind Turbines". PhD thesis. Aalborg Universitet, 2014.

[2] Shahrokh Akhlaghi, Ning Zhou, and Zhenyu Huang. "Adaptive Adjustment of Noise Covariance in Kalman Filter for Dynamic State Estimation". In: *IEEE Power & Energy Society General Meeting* (2017), pp. 1–5.

[3] Ali Almagbile, Jinling Wang, and Weidong Ding. "Evaluating the Performances of Adaptive Kalman Filter Methods in GPS/INS Integration". In: *Journal of Global Positioning Systems* 9.1 (2010), pp. 33–40. DOI: 10.5081/jgps.9.1.33.

[4] I. Arasaratnam and S. Haykin. "Square-Root Quadrature Kalman Filtering". In: *IEEE Transactions on Signal Processing* 56.6 (June 2008), pp. 2589–2593. DOI: 10.1109/TSP.2007.914964.

[5] I. Arasaratnam, S. Haykin, and R. J. Elliott. "Discrete-Time Nonlinear Filtering Algorithms Using Gauss-Hermite Quadrature". In: *Proceedings of the IEEE* 95.5 (May 2007), pp. 953–977. DOI: 10.1109/JPROC.2007.894705.

[6] Ienkaran Arasaratnam. "Cubature Kalman Filtering: Theory & Applications". PhD thesis. McMaster University, 2009.

[7] Ienkaran Arasaratnam and Simon Haykin. "Cubature Kalman Filters". In: *IEEE Transactions on Automatic Control* 54.6 (June 2009), pp. 1254–1269. DOI: 10.1109/TAC.2009.2019800.

[8] C. Bak, H.A. Madsen, and J. Johansen. "Influence from Blade-Tower Interaction on Fatigue Loads and Dynamics". In: *European conference, Wind Energy; Wind energy for the new millennium; 2001; Copenhagen*. 2001, pp. 394–397.

[9] Yaakov Bar-Shalom, X.-Rong Li, and Thiagalingam Kirubarajan. *Estimation with Applications To Tracking and Navigation*. John Wiley & Sons, 2001.

[10] Adi Ben-Israel and Thomas N.E. Greville. *Generalized Inverses: Theory and Applications (CMS Books in Mathematics)*. Springer, 2003.

[11] Jonathan C. Berg and A. Keith Miller. "Force Estimation via Kalman Filtering for Wind Turbine Blade Control". In: *IMAC-XXVIII*. 2010.

[12] M. Bertelè et al. "Wind inflow observation from load harmonics". In: *Wind Energy Science* 2.2 (2017), pp. 615–640. DOI: 10.5194/wes-2-615-2017.

[13] Gildas Besançon. *Nonlinear Observers and Applications*. Springer, 2007.

[14] Albert Betz. "Das Maximum der theoretisch möglichen Ausnützung des Windes durch Windmotoren". In: *Zeitschrift für das gesamte Turbinenwesen* 26.8 (1920), pp. 307–309.

[15] Albert Betz. *Windenergie und ihre Ausnützung durch Windmühlen*. Vandenhoeck & Ruprecht, 1926.

[16] Shibashis Bhowmik and René Spée. "Wind speed estimation based variable speed wind power generation". In: *Industrial Electronics Society* 2 (1998), pp. 596–601.

[17] Fernando D. Bianchi, Hernán De Battista, and Ricardo J. Mantz. *Wind Turbine Control Systems - Principles, Modelling and Gain Scheduling Design*. 2007.

[18] Gerald Bierman. *Factorization Methods for Discrete Sequential Estimation*. Ed. by Gerald Bierman. Academic Press, 1977.

[19] E. Bossanyi. "Developments in closed loop controller design for wind turbines". In: *2000 ASME Wind Energy Symposium*. Aerospace Sciences Meetings. 0. American Institute of Aeronautics and Astronautics, Jan. 2000. DOI: 10.2514/6.2000-27.

[20] E. Bossanyi et al. "Advanced controller research for multi-MW wind turbines in the UPWIND project". In: *Wind Energy* 15 (2012), pp. 119–145.

[21] E. A. Bossanyi. "Individual Blade Pitch Control for Load Reduction". In: *Wind Energy J.* 6 (2003), pp. 119–128.

[22] E. A. Bossanyi. "The Design of Closed Loop Controllers for Wind Turbines". In: *Wind Energy J* 3 (2000), pp. 149–163.

[23] E.A. Bossanyi, G. Ramtharan, and B. Savini. "The importance of control in wind turbine design and loading". In: *IEEE Control and Automation* (2009).

[24] C. L. Bottasso and A. Croce. *Advanced Control Laws for Variable-Speed Wind Turbines and Supporting Enabling Technologies*. Tech. rep. DIA-SR 09-01. Dipartimento di Ingegneria Aerospaziale, Politecnico di Milano, Jan. 2009.

[25] Carlo Luigi Bottasso and Alessandro Croce. *Cascading Kalman Observers of Structural Flexible and Wind States for Wind Turbine Control*. Tech. rep. Dipartimento di Ingegneria Aerospaziale, Politecnico di Milano, 2009.

[26] C.L. Bottasso, S. Cacciola, and J. Schreiber. "Local wind speed estimation, with application to wake impingement detection". In: *Renewable Energy* 116 (2018), pp. 155 –168. DOI: 10.1016/j.renene.2017.09.044.

[27] C.L. Bottasso and C.E.D. Riboldi. "Estimation of wind misalignment and vertical shear from blade loads". In: *Renewable Energy* 62 (2014), pp. 293 –302. DOI: 10.1016/j.renene.2013.07.021.

[28] C.L. Bottasso et al. "Spatial estimation of wind states from the aeroelastic response of a wind turbine". In: *The Science of Making Torque from Wind* (2010).

[29] Boubekeur Boukhezzar and Houria Siguerdidjane. "Nonlinear Control of Variable Speed Wind Turbines without wind speed measurement". In: *Proceedings of the 44th IEEE Conference on Decision and Control, and the European Control Conference* (2005).

[30] Dimitrios Bourlis. "Control Algorithms and Implementation for Variable Speed Stall Regulated Wind Turbines". PhD thesis. University of Leicester, 2010.

[31] Dimitris Bourlis and J.A.M. Bleijs. "A Wind Speed Estimation Method Using Adaptive Kalman Filtering For A Variable Speed Stall Regulated Wind Turbine". In: *Probabilistic Methods Applied to Power Systems (PMAPS), 2010 IEEE 11th International Conference on*. 2010. DOI: 10.1109/pmaps.2010.5528980.

[32] Karl Brammer and Gerhard Siffling. *Kalman-Bucy Filters*. Artech House, Inc., 1989.

[33] Steven D. Brown and Sarah C. Rutan. "Adaptive Kalman Filtering". In: *Journal of Research of the National Bureau of Standards* 90partic.6 (1985), pp. 403–407.

[34] Tony Burton et al. *Wind Energy Handbook*. 2. John Wiley and Sons, 2011.

[35] Franz D. Busse. "Precise formation-state estimation in low earth orbit using carrier differential GPS". PhD thesis. Stanford University, Mar. 2003.

[36] Franz D. Busse, Jonathan P. How, and James Simpson. "Demonstration of Adaptive Extended Kalman Filter for Low-Earth-Orbit Formation Estimation Using CDGPS". In: *Navigation* 50.2 (2003), pp. 79–93. DOI: 10.1002/j.2161-4296.2003.tb00320.x.

[37] S. Cacciola, Martha Bertelè, and C. L. Bottasso. "Simultaneous observation of wind shears and misalignments from rotor loads". In: *Journal of Physics: Conference Series* 753.052002 (2016). DOI: 10.1088/1742-6596/753/5/052002.

[38] Guanrong Chen. *Approximate Kalman Filtering*. World Scientific Publishing, 1993.

[39] Kuilin Chen and Jie Yu. "Short-term wind speed prediction using an unscented Kalman filter based state-space support vector regression approach". In: *Applied Energy* 113 (2014), pp. 690–705.

[40] Zhaozhong Chen et al. "Weak in the NEES?: Auto-tuning Kalman Filters with Bayesian Optimization". In: *CoRR* abs/1807.08855 (2018).

[41] Charles K. Chui and Guanrong Chen. *Kalman Filtering with Real-time Applications*. 4th ed. Springer, 2009.

[42] Robert F. Coleman and Arnold M. Feingold. *Theory of self-excited mechanical oscillations of helicopter rotors with hinged blades*. Tech. rep. National Advisory Committee for Aeronautics, 1958.

[43] Fred Daum. "Nonlinear Filters: Beyond the Kalman Filter". In: *IEEE Aerospace and Electronic Systems Magazine* 20.8 (Aug. 2005), pp. 57–69.

[44] C. Devriendt et al. "Damping estimation of an offshore wind turbine on a monopile foundation". In: *IET Renewable Power Generation* 7.4 (July 2013), pp. 401–412. DOI: 10.1049/iet-rpg.2012.0276.

[45] J. Dobbin et al. "Fully Integrated Design: Lifetime Cost of Energy Reduction for Offshore Wind". In: *Proceedings of the Twenty-fourth (2014), International Ocean and Polar Engineering Conference*. Vol. 3. 2014, pp. 415–423.

[46] Julian Ehlers and Amadou Diop. "Sensor Selection and State Estimation for Wind Turbine Controls". In: *45th AIAA Aerospace Sciences Meeting and Exhibit* (2007).

[47] Céline Faudot and Ole G. Dahlhaug. “Tidal Turbine Blades: Design and Dynamic Loads Estimation Using CFD and Blade Element Momentum Theory”. In: *ASME 2011 30th International Conference on Ocean, Offshore and Arctic Engineering, Rotterdam, The Netherlands*. 2011. DOI: 10.1115/OMAE2011-49740.

[48] Eckhard Gauterin et al. “Effective wind speed estimation: Comparison between Kalman Filter and Takagi-Sugeno observer techniques”. In: *{ISA} Transactions* 62 (2016). SI: Control of Renewable Energy Systems, pp. 60 –72. DOI: 10.1016/j.isatra.2015.11.016.

[49] D. Geffen et al. “Observability Based Parameter Identifiability for Biochemical Reaction Networks”. In: *2008 American Control Conference* (2008). DOI: 10.1109/ACC.2008.4586807.

[50] Sören Georg, Matthias Müller, and Horst Schulte. “Wind Turbine Model and Observer in Takagi-Sugeno Model Structure”. In: *Journal of Physics: Conference Series* 555.1 (2014), p. 012042.

[51] Martin Geyler and Boris Jasniewicz. “Parameter estimation for control design models based on operational modal analysis techniques”. In: *DEWEK 2010* (2010).

[52] A Giyanani, W A A M Bierbooms, and G J W van Bussel. “Estimation of rotor effective wind speeds using autoregressive models on Lidar data”. In: *Journal of Physics: Conference Series* 753.7 (2016), p. 072018.

[53] Thomas Glotzbach, Naveena Crasta, and Christoph Ament. “Observability Analyses and Trajectory Planning for Tracking of an Underwater Robot using Empirical Gramians”. In: *19th World Congress - The International Federation of Automatic Control* (2014).

[54] Gene H. Golub and Charles F. Van Loan. *Matrix computations*. 3rd ed. The Johns Hopkins University Press, 1996.

[55] C. Goodall and N. El-Sheimy. “Intelligent Tuning of a Kalman Filter using Low-Cost MEMS Inertial Sensors”. In: *Proceedings of 5th InternationalSymposium on Mobile Mapping Technology (MMT07)*. Padua. Italy, 2007, pp. 1–8.

[56] Felix Govaers, Alexander Charlish, and Wolfgang Koch. “Covariance Debiasing for the Distributed Kalman Filter”. In: *Proceedings of the 16th International Conference on Information Fusion, FUSION 2013*. July 2013.

[57] Mohinder S. Grewal and Angus P. Andrews. “Application of Kalman Filtering in Aerospace 1960 to the Present”. In: *IEEE* (2010).

[58] Mohinder S. Grewal and Angus P. Andrews. *Kalman Filtering: Theory and Practice. Using Matlab*. John Wiley & Sons, Inc., 2001.

[59] Mohinder S. Grewal and Angus P. Andrews. *Kalman Filtering. Theory and Practice Using Matlab*. 3rd ed. John Wiley & Sons, 2008.

[60] Sébastien Gros and Axel Schild. "Real-time economic nonlinear model predictive control for wind turbine control". In: *International Journal of Control* 90.12 (2017), pp. 2799–2812. DOI: 10.1080/00207179.2016.1266514.

[61] Sébastien Gros, Milan Vukov, and Moritz Diehl. "A Real-time MHE and NMPC Scheme for Wind Turbine Control". In: *IEEE Conference on Decision and Control* (2013).

[62] Peng Guo and David Infield. "Wind Turbine Tower Vibration Modeling and Monitoring by the Nonlinear State Estimation Technique (NSET)". In: *Energies* 5 (2012).

[63] F. Gustafsson and G. Hendeby. "Some Relations Between Extended and Unscented Kalman Filters". In: *IEEE Transactions on Signal Processing* 60.2 (Feb. 2012), pp. 545–555. DOI: 10.1109/TSP.2011.2172431.

[64] Frederik Gustafsson. *Adaptive Filtering and Change Detection*. John Wiley & Sons, LTD, 2000.

[65] Jianda Han, Qi Song, and Yuqing He. *Adaptive unscented Kalman filter and its applications in nonlinear control*. Ed. by Victor M. Moreno and Alberto Pigazo. INTECH Open Access Publisher, Apr. 2009. DOI: 10.5772/6799.

[66] Peter D. Hanlon. "Practical Implementation of Multiple Model Adaptive Estimation using Neyman-Pearson based hypothesis testing and spectral estimation tools". PhD thesis. Air Force Institute of Technology, Air University, Sept. 1996.

[67] Peter D. Hanlon and Peter S. Maybeck. "Multiple-model adaptive estimation using a residual correlation Kalman filter bank". In: *IEEE Transactions on Aerospace and Electronic Systems* 36.2 (Apr. 2000), pp. 393–406. DOI: 10.1109/7.845216.

[68] A.C. Hansen and C.P. Butterfield. "Aerodynamics of horizontal-axis wind turbines". In: *Annual Review of Fluid Mechanics* 25 (Jan. 1993), pp. 115–149. DOI: 10.1146/annurev.fl.25.010193.000555.

[69] M. H. Hansen et al. "Two Methods for Estimating Aeroelastic Damping of Operational Wind Turbine Modes from Experiments". In: *WIND ENERGY* 9 (2006), pp. 179–191. DOI: 10.1002/we.187.

[70] Martin O. L. Hansen. *Aerodynamics of Windturbines*. 2. Edition. Earthscan, London, 2008.

[71] Jouni Hartikainen, Arno Solin, and Simo Särkkä. *Optimal Filtering with Kalman Filters and Smoothers a Manual for the Matlab toolbox EKF/UKF Version 1.3*. http://becs.aalto.fi/en/research/bayes/ekfukf/. Aalto University. 2011.

[72] Erich Hau. *Wind Turbines. Fundamentals, Technologies, Application, Economics*. 2. Edition. Springer-Verlag Berlin Heidelberg, 2006.

[73] Melanie Hau. *Promising Load Estimation Methodologies for Wind Turbine Components (Upwind)*. Tech. rep. 55 pages. Institut für Solare Energieversorgungstechnik (ISET), 2008.

[74] Anton J. Haug. *Bayesian Estimation and Tracking: A Practical Guide*. John Wiley & Sons, Inc., 2012.

[75] Felix Heß and Georg Seyboth. "Individual Pitch Control with Tower Side-to-Side Damping". In: *Dewek 2010*. 2010.

[76] L. C. Henriksen, M. H. Hansen, and M. K. Poulsen. "A simplified dynamic inflow model and its effect on the performance of free mean wind speed estimation". In: *Wind Energy* 16 (2013), pp. 1213–1224.

[77] Christian Himpe. *The Empirical Gramian Framework*. Tech. rep. Computational Methods in Systems and Control Theory Group at the Max Planck Institute for Dynamics of Complex Technical Systems, 2016.

[78] Christian Himpe and Mario Ohlberger. "A Unified Software Framework for Empirical Gramians". In: *Journal of Mathematics* 2013.Article ID 365909 (2013), pp. 1–6. DOI: 10.1155/2013/365909.

[79] E.L. van der Hooft and T.G. van Engelen. "Estimated Wind Speed Feed Forward Control for Wind Turbine Operation Optimisation". In: *EWEC* (2004).

[80] E.L. van der Hooft and T.G. van Engelen. *Feed forward control of estimated wind speed*. Tech. rep. ECN, 2003.

[81] John Hubbard, Dierk Schleicher, and Scott Sutherland. "How to find all roots of complex polynomials by Newton's method". In: *Inventiones mathematicae* 146.1 (Oct. 2001), pp. 1–33. DOI: 10.1007/s002220100149.

[82] Mikel Iribas and Ioan Doré Landau. "Closed Loop Identification of Wind Turbines Models for Pitch Control". In: *7th Mediterranean Conference on Control and Automation (MED'09)*. 2009.

[83] M. Iribas-Latour and I-D. Landau. "Identification in closed-loop operation of models for collective pitch robust controller design". In: *Wind Energy* 16.3 (2012), pp. 383–399. DOI: 10.1002/we.1494.

[84] Rolf Isermann and Marco Münchhof. *Identification of Dynamic Systems. An Introduction with Applications*. 1st ed. Springer, 2011. DOI: 10.1007/978-3-540-78879-9.

[85] Ivana Ivanović, Mario Barišić, and Zlatka Tečec Ribarić. "Implementing the wind speed estimation algorithm in the target embedded control system". In: *International Conference on ELECTRICAL DRIVES and POWER ELECTRONICS - EDPE* (2013).

[86] Boris Jasniewicz. *Online estimation of mechanical loads for wind turbines (Upwind)*. Tech. rep. 27 pages. Fraunhofer IWES, 2011.

[87] Boris Jasniewicz and Martin Geyler. "Wind turbine modelling and identification for control system applications". In: *Fraunhofer Institute for Wind Energy and Energy System Technology (IWES), Division Control Engineering and Energy Storages* (2010).

[88] M. Jelavić, V. Petrović, and N. Perić. "Individual pitch control of wind turbine based on loads estimation". In: *2008 34th Annual Conference of IEEE Industrial Electronics*. Nov. 2008, pp. 228–234. DOI: 10.1109/IECON.2008.4757957.

[89] Mate Jelavić, Vlaho Petrović, and Nedjeljko Perić. "Estimation based Individual Pitch Control of Wind Turbine". In: *ATKAFF* 51.2 (2010), pp. 181–192.

[90] Debashisha Jena and Saravanakumar Rajendran. "A review of estimation of effective wind speed based control of wind turbines". In: *Renewable and Sustainable Energy Reviews* 43.C (2015), pp. 1046–1062. DOI: 10.1016/j.rser.2014.11.08.

[91] Peter Hjuler Jensen, Takis Chaviaropoulos, and Anand Natarajan. *LCOE reduction for the next generation offshore wind turbines. Outcomes from the INNWIND.EU Project*. Tech. rep. DTU and NTUA, 2017.

[92] Manli Jiang and Peng Guo. "Wind turbine performance monitoring based on nonlinear state estimate technique". In: *2016 Chinese Control and Decision Conference (CCDC)*. May 2016, pp. 4662–4666. DOI: 10.1109/CCDC.2016.7531825.

[93] J. Andrew Johnson. *Error Ellipse Computation*. Tech. rep. 2004.

[94] B.J. Jonkman and L. Kilcher. *TurbSim User"s Guide: Version 1.06.00*. Tech. rep. NREL, 2012.

[95] J. Jonkman et al. *Definition of a 5-MW Reference Wind Turbine for Offshore System Development*. Tech. rep. Technical Report NREL/TP-500-38060 February 2009. National Renewable Energy Laboratory (NREL), 2009.

[96] Jason Jonkman and Bonnie Jonkman. *NWTC Information Portal (FAST v8)*. 2015. URL: https://nwtc.nrel.gov/FAST8.

[97] Jason M. Jonkman. "Dynamics Modeling and Loads Analysis of an Offshore Floating Wind Turbine". Technical Report NREL/TP-500-41958. National Wind Technology Center, Oct. 2007.

[98] N.E. Joukowsky. "Windmill of the NEJ type". In: *Transactions of the Central Institute for Aero-hydrodynamics of Moscow* 1.57 (1920).

[99] Simon Julier and Jeffrey K. Uhlmann. *A General Method for Approximating Nonlinear Transformations of Probability Distributions*. Tech. rep. 1996.

[100] Simon J. Julier. "The Spherical Simplex Unscented Transformation". In: *Proceedings of the 2003 IEEE - American Control Conference* 3 (2003), pp. 2430–2434. DOI: 10.1109/ACC.2003.1243439.

[101] Simon J. Julier and Jeffrey K. Uhlmann. "A New Extension of the Kalman Filter to Nonlinear Systems". In: *11th International symposium on Aerospace/Defence Sensing, Simulation and Control* (1997).

[102] Simon J. Julier and Jeffrey K. Uhlmann. "Unscented Filtering and Nonlinear Estimation". In: *Proceedings of the IEEE* 92.3 (Mar. 2004), pp. 401–423.

[103] R.E. Kalman and R.S. Bucy. "New Results in Linear Filtering and Prediction Theory". In: *Trans. ASME - Journal of Basic Engineering* 83 (1961), pp. 95–108.

[104] Rudolf E. Kalman. “A New Approach to Linear Filtering and Prediction Problems”. In: *Trans. ASME - Journal of Basic Engineering* 82.D (1960), pp. 35–45.

[105] N. Kavitha and S. Vijayachitra. “Extended Kalman Filter based State Estimation of Wind Turbine”. In: *International Journal of Engineering Trends and Technology (IJETT)* 5.4 (Nov. 2013).

[106] M. Najafi Khoshrodi, Mohammad Jannati, and Tole Sutikno. “A Review of Wind Speed Estimation for Wind Turbine Systems Based on Kalman Filter Technique”. In: *International Journal of Electrical and Computer Engineering (IJECE)* 6.4 (Aug. 2016), pp. 1406–1411. DOI: 10.11591/ijece.v6i4.10735.

[107] Torben Knudsen. *Wind turbine state estimation*. Tech. rep. The Faculty of Engineering and Science Department of Electronic Systems Automation & Control, 2014.

[108] Torben Knudsen and Thomas Bak. “Simple Model for Describing and Estimating Wind Turbine Dynamic Inflow”. In: *American Control Conference* (2013).

[109] Torben Knudsen, Thomas Bak, and Mohsen Soltani. “Prediction models for wind speed at turbine locations in a wind farm”. In: *Wind Energy* 14 (2011), pp. 877–894. DOI: 10.1002/we.491.

[110] Arne Körber. “Extreme and Fatigue Load Reducing Control for Wind Turbines: A Model Predictive Control Approach using Robust State Constraints”. PhD thesis. Technischen Universität Berlin, 2014. DOI: 10.14279/depositonce-4133.

[111] A. Kumar and K. Stol. “Simulating Feedback Linearization control of wind turbines using high-order models”. In: *Wind Energy* 13.5 (2009), pp. 419–432. DOI: 10.1002/we.363.

[112] Avishek A. Kumar, Patrick J. Rainey, and Ervin A. Bossanyi. “LIDAR Assisted Model Predictive Control of a Next Generation Wind Turbine for Tower Fatigue Load Reduction and Improved Speed Control”. In: *EWEC* (2015).

[113] F. W. Lanchester. “A Contribution to the Theory of Propulsion and the Screw Propeller”. In: *Journal of the American Society for Naval Engineers* 27.2 (1915), pp. 509–510. DOI: 10.1111/j.1559-3584.1915.tb00408.x.

[114] T. J. Larsen and T. D. Hanson. “A method to avoid negative damped low frequent tower vibrations for a floating, pitch controlled wind turbine”. In: *The Science of Making Torque from Wind* (2007).

[115] *Lazard’s Levelized Cost of Energy Analysis - Version 12.0*. Nov. 2018.

[116] Deok-Jin Lee and Kyle T. Alfriend. “Adaptive Sigma Point Filtering for State and Parameter Estimation”. In: *AIAA/AAS Astrodynamics Specialist Conference and Exhibit* (2004).

[117] Tine Lefebvre, Herman Bruyninckx, and Joris De Schutter. “Kalman filters for non-linear systems: a comparison of performance”. In: *International Journal of Control* 77.7 (2004), pp. 639–653. DOI: 10.1080/00207170410001704998.

[118] X. R. Li and Y. Bar-Shalom. "A recursive multiple model approach to noise identification". In: *IEEE Transactions on Aerospace and Electronic Systems* 30.3 (July 1994), pp. 671–684. DOI: 10.1109/7.303738.

[119] X Rong Li, Zhanlue Zhao, and Vesselin P. Jilkov. "Practical Measures and Test for Credibility of an Estimator". In: *Proc. Workshop on Estimation, Tracking, and Fusion - A Tribute to Yaakov Bar-Shalom*. Jan. 2001, pp. 481–495.

[120] Martin E. Liggins, David L. Hall, and James Llinas, eds. *Handbook of Multisensor Data Fusion. Theory and Practice*. 2nd ed. CRC Press Taylor & Francis Group, 2009.

[121] Wai Hou Lio. "Blade-pitch Control for Wind Turbine Load Reductions". PhD thesis. The University of Sheffield, 2017.

[122] Lennart Ljung. *System Identification: Theory for the User*. 2. Prentice Hall, 1999.

[123] Xin Ma, Niels K. Poulsen, and Henrik Bindner. *Estimation of Wind Speed in Connection to a Wind Turbine*. Tech. rep. Technical University of Denmark, 1995.

[124] D. Magill. "Optimal adaptive estimation of sampled stochastic processes". In: *IEEE Transactions on Automatic Control* 10.4 (Oct. 1965), pp. 434–439. DOI: 10.1109/TAC.1965.1098191.

[125] M. Majeed and Indra Narayan Kar. "Aerodynamic parameter estimation using adaptive unscented Kalman filter". In: *Aircraft Engineering and Aerospace Technology* 85.4 (2013), pp. 267–279. DOI: 10.1108/AEAT-Mar-2011-0038.

[126] Petar Mateljak, Vlaho Petrović, and Mato Baotić. "Dual Kalman Estimation of Wind Turbine States and Parameters". In: *18th International Conference on Process Control* (2011). Slovak University of Technology in Bratislava Institute of Information Engineering, Automation, and Mathematics.

[127] Peter S. Maybeck. *Stochastic models, estimation, and control*. Vol. 1. Mathematics in Science and Engineering. Academic Press, Inc., 1979.

[128] Peter S. Maybeck. *Stochastic models, estimation, and control*. Vol. 2. Academic Press, Inc., 1982.

[129] R. Mehra. "On-line identification of linear dynamic systems with applications to Kalman filtering". In: *IEEE Transactions on Automatic Control* 16.1 (Feb. 1971), pp. 12–21. DOI: 10.1109/TAC.1971.1099621.

[130] Raman K. Mehra. "Approaches to adaptive filtering". In: *IEEE Transactions on Automatic Control* 17.5 (Oct. 1972), pp. 693–698. DOI: 10.1109/TAC.1972.1100100.

[131] Raman K. Mehra. "On the identification of variances and adaptive Kalman filtering". In: *IEEE Trans. Autom. Control* 15.2 (1970), pp. 175–184. DOI: 10.1109/TAC.1970.1099422.

[132] Rudolph van der Merwe. “Sigma-Point Kalman Filters for Probabilistic Inference in Dynamic State-Space Models”. PhD thesis. OGI School of Science & Engineering at Oregon Health & Science University, 2004.

[133] Rudolph van der Merwe and Eric A. Wan. “Efficient Derivative-Free Kalman Filters for Online Learning”. In: *ESANN'2001 proceedings - European Symposium on Artificial Neural Networks* (2001), pp. 205–210.

[134] Rudolph van der Merwe and Eric A. Wan. “The square-root Unscented Kalman Filter for state and parameter-estimation”. In: *IEEE International Conference on Acoustics, Speech, and Signal Processing, 2001. Proceedings. (ICASSP '01). 2001* 6 (2001), pp. 3461–3464.

[135] Manfred Mevenkamp and Josef Petschenka. “Regelung und Zustandsüberwachung von Windkraftanlagen - Methoden des Rapid Prototyping”. In: *Kasseler Symposium Energie-Systemtechnik '98*. 1998.

[136] A. H. Mohamed and K. P. Schwarz. “Adaptive Kalman Filtering for INS/GPS”. In: *Journal of Geodesy* 73.4 (1999), pp. 193–203. DOI: 10.1007/s001900050236.

[137] C. Moné et al. *2013 Cost of Wind Energy Review*. Tech. rep. NREL/TP-5000-63267. National Renewable Energy Laboratory, Feb. 2015.

[138] P.J. Moriarty and A.C. Hansen. *AeroDyn Theory Manual*. Tech. rep. NREL & Windward Engineering, 2005.

[139] Iulian Munteanu and Gildas Besançon. “Control-based strategy for effective wind speed estimation in wind turbines”. In: *IFAC Proceedings Volumes* 47.3 (2014). 19th IFAC World Congress, pp. 6776 –6781. DOI: 10.3182/20140824-6-ZA-1003.01952.

[140] Iulian Munteanu et al. *Optimal Control of Wind Energy Systems - Towards a Global Approach*. 2008.

[141] Kenneth A. Myers and Byron D. Tapley. “Adaptive Sequential Estimation with Unknown Noise Statistics”. In: *IEEE Transactions on Automatic Control* 21.4 (1976), pp. 520–523. DOI: 10.1109/TAC.1976.1101260.

[142] Jorge Nocedal and Stephen J. Wright. *Numerical Optimization*. Ed. by Thomas V. Mikosch, Stephen M. Robinson, and Sidney I. Resnick. 2nd ed. Springer, 2006.

[143] Magnus Nørgaard and Niels K. Poulsen. *Advances in Derivative-Free State Estimation for Nonlinear Systems*. Tech. rep. Department of Mathematical Modeling, Technical University of Denmark, 2004.

[144] Magnus Nørgaard, Niels K. Poulsen, and Ole Ravn. “New developments in state estimation for nonlinear systems”. In: *Automatica* 36.11 (Nov. 2000), pp. 1627–1638.

[145] P. Novak et al. “Modeling and Control of Variable-Speed Wind-Turbine Drive-System Dynamics”. In: *IEEE Control Systems* (Aug. 1995).

[146] Peter F. Odgaard, Chris Damgaard, and Rasmus Nielsen. "On-Line Estimation of Wind Turbine Power Coefficients Using Unknown Input Observers". In: *IFAC Proceedings Volumes* 41.2 (2008). 17th IFAC World Congress, pp. 10646 –10651. DOI: 10.3182/20080706-5-KR-1001.01804.

[147] Peter Fogh Odgaard and Jakob Stoustrup. "Unknown Input Observer Based Scheme for Detecting Faults in a Wind Turbine Converter". In: *IFAC Proceedings Volumes* 42.8 (2009). 7th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes, pp. 161 –166. DOI: 10.3182/20090630-4-ES-2003.00027.

[148] Valery L. Okulov and Gijs A.M. van Kuik. "The Betz–Joukowsky limit: on the contribution to rotor aerodynamics by the British, German and Russian scientific schools". In: *Wind Energy* 15.2 (2011), pp. 335–344. DOI: 10.1002/we.464.

[149] Romeo Ortega, Fernando Mancilla-David, and Fernando Jaramillo. "A globally convergent wind speed estimator for wind turbine systems". In: *International Journal of Adaptive Control and Signal Processing* 27.5 (2013), pp. 413–425. DOI: 10.1002/acs.2319.

[150] Romeo Ortega, Fernando Mancilla-David, and Fernando Jaramillo. "A Globally Convergent Wind Speed Estimator for Windmill Systems". In: *IEEE Conference on Decision and Control and European Control Conference (CDC-ECC)* (2011).

[151] K Z Østergaard, P Brath, and J Stoustrup. "Estimation of effective wind speed". In: *Journal of Physics: Conference Series 75 (2007). The Science of Making Torque from Wind* 75 (2007).

[152] Kasper Zinck Østergaard. "Robust, Gain-Scheduled Control of Wind Turbines". Load, Aerodynamics and Control Vestas Wind Systems A/S. PhD thesis. Automation and Control Department of Electronic Systems Aalborg University, 2008.

[153] M. Oussalah and J. De Schutter. "Adaptive Kalman Filter for Noise Identification". In: *International Conference on Noise and Vibration Engineering (ISMA25)*. 2000, pp. 1225–1232.

[154] Robert H. Park. "Two-reaction theory of synchronous machines". In: *Transactions of the AIEE* 48 (1929).

[155] V. Petrović, M. Jelavić, and N. Perić. "Identification of wind turbine model for individual pitch controller design". In: *2008 43rd International Universities Power Engineering Conference*. Sept. 2008, pp. 1–5. DOI: 10.1109/UPEC.2008.4651585.

[156] V. Petrović and N. Perić. "Identification of wind turbine mathematical model suitable for advanced controller design". In: *2012 Proceedings of the 35th International Convention MIPRO*. May 2012, pp. 835–840.

[157] Raymond Pierrehumbert. "There is no Plan B for dealing with the climate crisis". In: *Bulletin of the Atomic Scientists* 75.5 (2019), pp. 215–221. DOI: 10.1080/00963402.2019.1654255.

[158] Jerson Rogério Pinheiro Vaz, João Tavares Pinho, and André Luiz Amarante Mesquita. "An extension of BEM method applied to horizontal-axis wind turbine design". In: *Renewable Energy* 36.6 (2011), pp. 1734 –1740. DOI: https://doi.org/10.1016/j.renene.2010.11.018.

[159] Florian A. Potra and Stephen J. Wright. "Interior-point methods". In: *Journal of Computational and Applied Mathematics* 124.1 (2000). Numerical Analysis 2000. Vol. IV: Optimization and Nonlinear Equations, pp. 281 –302. DOI: 10.1016/S0377-0427(00)00433-7.

[160] Thomas D. Powell. "Automated Tuning of an Extended Kalman Filter Using the Downhill Simplex Algorithm". In: *Journal of Guidance, Control and Dynamics* 25.5 (Oct. 2002), pp. 901–908. DOI: 10.2514/2.4983.

[161] William H. Press et al. *Numerical recipes: The art of scientific computing*. 3rd ed. Cambridge University Press, Cambridge, UK, 2007.

[162] Juntong Qi, Jianda Han, and Zhenwei Wu. "Rotorcraft UAV actuator failure estimation with KF-based adaptive UKF algorithm". In: *2008 American Control Conference*. June 2008, pp. 1618–1623. DOI: 10.1109/ACC.2008.4586723.

[163] Jesse L. Reynolds and Gernot Wagner. "Highly decentralized solar geoengineering". In: *Environmental Politics* 0.0 (2019), pp. 1–17. DOI: 10.1080/09644016.2019.1648169.

[164] Carlo E.D. Riboldi. "Advanced Control Laws for Variable-Speed Wind Turbines and Supporting Enabling Technologies". PhD thesis. Politecnico di Milano, Department of Aerospace Engineering, 2012.

[165] B Ritter, C Schmitz, and PF Pelz. "Control-oriented modelling and state estimation of tidal turbines with pitch control". In: *20th International Seminar on Hydropower Plants*. Nov. 2018.

[166] B Ritter et al. "Adaptive Sigma-Point Kalman Filtering for Wind Turbine State and Process Noise Estimation". In: *Journal of Physics: Conference Series* 1037.032003 (June 2018), pp. 1–11. DOI: 10.1088/1742-6596/1037/3/032003.

[167] Bastian Ritter, Niko Chrysalidis, and Ulrich Konigorski. "Application of Nonlinear Kalman Filters to Wind Turbine State Observation". In: *Proceedings of the 12th EAWE PhD Seminar on Wind Energy in Europe* (2016).

[168] Bastian Ritter and Ulrich Konigorski. "Advanced Multivariable Control Design for Modern Multi-MW Wind Turbines". In: *Proceedings of the 11th EAWE PhD Seminar on Wind Energy in Europe* (2015).

[169] Bastian Ritter, Edwin Mora, and Axel Schild. "Observability Analysis for Horizontal Axis Wind Turbines using Empirical Gramian Matrices". In: *Journal of Physics: Conference Series* 1102.012021 (Oct. 2018), pp. 1–11. DOI: 10.1088/1742-6596/1102/1/012021.

[170] Bastian Ritter, Axel Schild, and Ulrich Konigorski. "Making nonlinear state estimation techniques ready for use in industrial wind turbine control systems". In: *Proc. of the WindEurope Summit 2016, Hamburg, Germany*. 2016.

[171] Bastian Ritter et al. "Adaptive Master-Slave Cubature Kalman Filters subject to State Inequality Constraints for Wind Turbine State Estimation". In: *2019 American Control Conference (ACC)*. July 2019, pp. 3482–3487.

[172] Bastian Ritter et al. "Multivariable Model for Simulation and Control Design of Wind Turbines". In: *Proceedings of the 12th German Wind Energy Conference (DEWEK)* (2015).

[173] Bastian Ritter et al. "Noise adaptive design of sigma-point Kalman filters for nonlinear wind turbine state estimation". In: *Wind Europe Conference & Exhibition 2017, Amsterdam, The Netherlands, November 28-30, 2017*. 2017.

[174] Bastian Ritter et al. "The design of nonlinear observers for wind turbine dynamic state and parameter estimation". In: *Journal of Physics: Conference Series* 753.052029 (2016), pp. 1–12. DOI: 10.1088/1742-6596/753/5/052029.

[175] Pedro Rosas. "Dynamic Influences of Wind Power on the Power System". PhD thesis. Technical University of Denmark (DTU), 2003.

[176] Jonathan Rose and Ian A. Hiskens. "Estimating wind turbine parameters and quantifying their effects on dynamic behavior". In: *2008 IEEE Power and Energy Society General Meeting - Conversion and Delivery of Electrical Energy in the 21st Century*. Ed. by IEEE. Institute of Electrical and Electronics Engineers (IEEE), July 2008. DOI: 10.1109/pes.2008.4596862.

[177] R.B. de Ruijter. "Controllability and observability of a 2D wind farm model". MA thesis. TU Delft, Delft Center for Systems and Control, 2017.

[178] Frank Sandner. "Reduced Model Design of a Floating Wind Turbine". MA thesis. University of Stuttgart, 2012.

[179] S. Särkkä and J. Hartikainen. "Sigma point methods in optimal smoothing of non-linear stochastic state space models". In: *2010 IEEE International Workshop on Machine Learning for Signal Processing*. Aug. 2010, pp. 184–189. DOI: 10.1109/MLSP.2010.5589160.

[180] Simo Särkkä. *Bayesian filtering and smoothing*. Vol. 3. Cambridge University Press, 2013.

[181] D. Sbarbaro and R. Pena. "A non-linear wind velocity observer for a small wind energy system". In: *Proceedings of the 39th IEEE Conference on Decision and Control (Cat. No.00CH37187)*. Vol. 4. 2000, 3086–3087 vol.4. DOI: 10.1109/CDC.2000.912169.

[182] T.S. Schei. "A finite-difference method for linearization in nonlinear estimation algorithms". In: *Automatica* 33.11 (Nov. 1997), pp. 2053–2058.

[183] Axel Schild, Sébastien Gros, and Bastian Ritter. "High performance nonlinear model predictive control for wind turbines". In: *Wind Europe Summit 2016*. 2016.

[184] Thorsten Schlicht. "Systematischer Entwurf adaptiver Kalman Filter für Windturbinen". MA thesis. Technical University of Darmstadt, 2017.

[185] David Schlipf. "Lidar-Assisted Control Concepts for Wind Turbines". PhD thesis. University of Stuttgart, 2016.

[186] David Schlipf and Steffen Raach. "Turbulent Extreme Event Simulations for Lidar-Assisted Wind Turbine Control". In: *Journal of Physics: Conference Series* 753.052011 (2016). DOI: 10.1088/1742-6596/753/5/052011.

[187] David Schlipf, Dominik Johannes Schlipf, and Martin Kühn. "Nonlinear model predictive control of wind turbines using LIDAR". In: *Wind Energy* 16 (2013), pp. 1107–1129. DOI: 10.1002/we.1533.

[188] Stanley F. Schmidt. "Application of State-Space Methods to Navigation Problems". In: ed. by C.T. Leondes. Vol. 3. Advances in Control Systems. Elsevier, 1966, pp. 293 –340. DOI: 10.1016/B978-1-4831-6716-9.50011-4.

[189] Joris De Schutter et al. *Kalman Filters: A Tutorial*. Tech. rep. Katholieke Universiteit Leuven, 1999.

[190] K. Selvam et al. "Feedback-feedforward individual pitch control for wind turbine load reduction". In: *INTERNATIONAL JOURNAL OF ROBUST AND NONLINEAR CONTROL* 19 (2009), pp. 72–91.

[191] Eric Simley and Lucy Y. Pao. "Evaluation of a wind speed estimator for effective hub-height and shear components". In: *Wind Energy* 19.1 (Jan. 2014), pp. 167–184.

[192] Eric J. Simley. "Wind Speed Preview Measurement and Estimation for Feedforward Control of Wind Turbines". PhD thesis. University of Colorado, 2015.

[193] Dan Simon. "Kalman Filtering". In: *Embedded Systems Programming* 14.6 (June 2001), pp. 72–79.

[194] Dan Simon. *Optimal State Estimation. Kalman, H_∞, and Nonlinear Approaches*. John Wiley & Sons, 2006.

[195] Abhay K. Singh and Juergen Hahn. "On the Use of Empirical Gramians for Controllability and Observability Analysis". In: *Proceedings of the 2005, American Control Conference, 2005.* (June 2005), pp. 140–146. DOI: 10.1109/ACC.2005.1469922.

[196] T.G. van Engelen S.K. Kanev. *Exploring the Limits in Individual Pitch Control*. Tech. rep. ECN, 2009.

[197] Sigurd Skogestad and Ian Postlethwaite. *Multivariable Feedback Control. Analysis and Design*. 1st ed. John Wiley & Sons, 2001.

[198] Mohsen Nourbakhsh Soltani et al. "Estimation of Rotor Effective Wind Speed: A Comparison". In: *IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY* 21.4 (July 2013), pp. 1155–1167.

[199] Q. Song and Y. He. "Adaptive unscented Kalman filter for estimation of modelling errors for helicopter". In: *2009 IEEE International Conference on Robotics and Biomimetics (ROBIO)*. Dec. 2009, pp. 2463–2467. DOI: 10.1109/ROBIO.2009.5420406.

[200] Q. Song, J. Qi, and J. Han. "An Adaptive UKF Algorithm and Its Application in Mobile Robot Control". In: *2006 IEEE International Conference on Robotics and Biomimetics*. Dec. 2006, pp. 1117–1122. DOI: 10.1109/ROBIO.2006.340085.

[201] Qi Song, Zhe Jiang, and Jianda Han. "Noise Covariance Identification Based Adaptive UKF with Application to Mobile Robot Systems". In: *2007 IEEE International Conference on Robotics and Automation*. 2007.

[202] H. W. Sorenson. "Least-squares estimation: from Gauss to Kalman". In: *IEEE Spectrum* 7 (July 1970), pp. 63–68.

[203] Robert F. Stengel. *Optimal Control and Estimation*. 2nd ed. Dover Edition (orig. pub. by John Wiley & Sons), 1994.

[204] Robert F. Stengel. *Optimal Control and Estimation*. 2012.

[205] Karl A. Stol and Lee J. Fingersh. *Wind Turbine Field Testing of State-Space Control Designs*. Tech. rep. University of Auckland and National Renewable Energy Laboratory, 2004.

[206] Gilbert Strang. *Linear Algebra and Its Applications*. 4th. Cengage Learning, 2006.

[207] P. Sudev, J. P. Anita, and P. Sudheesh. "Nonlinear state estimation of wind turbine". In: *2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*. Sept. 2017, pp. 354–358. DOI: 10.1109/ICACCI.2017.8125866.

[208] P. Towers and B. Ll. Jones. "Real-time wind field reconstruction from LiDAR measurements using a dynamic wind model and state estimation". In: *Wind Energy* 19.1 (Jan. 2014). The University of Sheffield, pp. 133–150.

[209] Jeffrey K. Uhlmann. "Dynamic Map Building and Localization: New Theoretical Foundations". PhD thesis. University of Oxford, 1995.

[210] Ranjan Vepa. "Ambulatory Position Tracking of Prosthetic Limbs Using Multiple Satellite Aided Inertial Sensors and Adaptive Mixing". In: *Journal of Navigation* 64.02 (Apr. 2011), pp. 295–310.

[211] Ranjan Vepa. "Control and Estimation of a Variable Pitch Wind Turbine for Maximum Power Point Tracking". In: *Wind Turbines - Design, Control and Applications*. Ed. by Abdel Ghani Aissaoui and Ahmed Tahour. Rijeka: InTech, 2016. Chap. 11. DOI: 10.5772/62723.

[212] Miroslav Šimandl. "Lecture notes on state estimation of nonlinear non-Gaussian stochastic systems". Department of Cybernetics, Faculty of Applied Science, University of West Bohemia in Pilsen. 2006.

[213] Eric A. Wan and Rudolph van der Merwe. "The Unscented Kalman Filter for Nonlinear Estimation". In: *IEEE Symposium 2000 (AS-SPCC)* (2000).

[214] Wout Weijtjens et al. "Automated operational modal analysis on an offshore wind turbine: Challenges, results and opportunities". In: *6th International Operational Modal Analysis Conference, IOMAC 2015*. May 2015.

[215] Greg Welch and Gary Bishop. *An Introduction to the Kalman Filter*. Tech. rep. SIGGRAPH 2001. University of North Carolina at Chapel Hill, USA, 2001.

[216] Greg Welch and Gary Bishop. *An Introduction to the Kalman Filter*. Tech. rep. University of North Carolina at Chapel Hill, USA, 2006.

[217] Nobert Wiener. *Extrapolation, Interpolation, and Smoothing of Stationary Time Series*. MIT Press Classic, 1949.

[218] A. D. Wright and L. J. Fingersh. *Advanced Control Design for Wind Turbines: Part I: Control Design, Implementation, and Initial Tests*. Tech. rep. NREL, 2008.

[219] A.D. Wright, L.J. Fingersh, and M.J. Balas. *Testing State-Space Controls for the Controls Advanced Research Turbine*. Tech. rep. NREL and University of Wyoming, 2006.

[220] Alan D. Wright. *Modern Control Design for Flexible Wind Turbines*. Tech. rep. NREL, 2004.

[221] Margaret H. Wright. "The interior-point revolution in optimization: History, recent developments, and lasting consequences". In: *Bull. Amer. Math. Soc.* 42.1 (Sept. 2015), pp. 39–56. DOI: 10.1090/S0273-0979-04-01040-7.

[222] Stephen J. Wright. *Primal-dual interior-point methods*. Society for Industrial and Applied Mathematics (SIAM), 1997. DOI: 10.1137/1.9781611971453.

[223] Yuanxin Wu et al. "Unscented Kalman Filtering for Additive Noise Case: Augmented versus Nonaugmented". In: *IEEE SIGNAL PROCESSING LETTERS* 12.5 (May 2005), p. 4.

[224] Renato Zanetti and Kyle J. DeMars. "Joseph Formulation of Unscented and Quadrature Filters with Application to Consider States". In: *Journal of Guidance Control and Dynamics* 36.6 (Nov. 2013), pp. 1860–1864. DOI: 10.2514/1.59935.

[225] Huo Zhihong, Zheng Yuan, and Xu Chang. "State estimation for wind turbine system based on Kalman filter". In: *2nd International Symposium on Systems and Control in Aerospace and Astronautics, 2008. ISSCAA 2008.* (2008).